NPS ARCHIVE
1997.09
LEBAS, P.

DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY CA 93943-5101

DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
411 DYER RD ROOM 100
MONTEREY CA 93943-5101

Maximizing AUV Slow Speed Performance

by

Phillip J. LeBas

B.S. Electrical Engineering
Auburn University, 1984

Submitted to the Joint Committee on Oceanographic Engineering in partial fulfillment of the requirements for the degree of

Master of Science
in Ocean Engineering
at the

Woods Hole Oceanographic Institution

and the

Massachusetts Institute of Technology

September 1997

© Phillip J. LeBas, 1997. All Rights Reserved.

The author hereby grants to MIT and WHOI permission to reproduce and distribute publicly paper and electronic copies of this thesis document in whole or in part.

nps Archive
1997.09
LeBas, P.

Maximizing AUV Slow Speed Performance
by

Phillip J. LeBas

Submitted to the Joint Committee on Oceanographic Engineering
on July 21, 1997, in partial fulfillment of the requirements for the
degree of Master of Science

ABSTRACT

A review of the Odyssey IIb autonomous underwater vehicle shows that energy costs associated with vehicle controls can be reduced and operational flexibility improved with relatively simple, low cost improvements. Because the operating speed that minimizes forward drag is not necessarily the same as that required for optimum sensor performance, a variable speed capability extending to the bottom of the vehicle speed range is sought. Optimizing Odyssey IIB AUV performance for slower speed operations and extended duration missions necessitates a multi-disciplinary review including control system design, hydrodynamic performance and sensor selection and utilization. Reducing the vehicle controls-fixed directional instability by adding vertical fixed fins, implementing an actuation filter, and designing a model based adaptive sliding controller improves the variable speed performance and reduces the control actuation necessary to provide the desired performance level with energy savings.

Thesis Supervisor: Dr. Dana R. Yoerger

Title: Associate Scientist
Applied Ocean Physics and Engineering,
Woods Hole Oceanographic Institution

Table of Contents

Chapter 1 Thesis Description 7

1.1 Motivation 7
1.2 Research Objectives 12
1.3 Thesis Outline 13

Chapter 2 AUV Dynamics 14

2.1 Equations of Motion-Standard Submarine Equations 14
2.1.1 Coordinate System 14
2.1.2 Coordinate Transformation 16
2.1.3 Weight and Buoyancy 17
2.2 Linearized Dynamics 18
2.3 Weight and Buoyancy 20
2.4 Body and Fin Fluid Forces and Moments 21
2.5 Linearized Model for Control System Design 22
2.6 Models for Pitch and Yaw 25
2.6.1 Model for Pitch 26
2.6.2 Model for Yaw 26
2.7 Summary 27

Chapter 3 Control System Design 28

3.1 Robust Sliding Control Overview 28
3.2 Controller Design 31
3.2.1 Robust Sliding Controller Design for Pitch 31
3.2.2 Adaptive Controller Design for Pitch 35
3.2.3 Limiting adaptation 37
3.3 Depth 38
3.4 Yaw 40
3.5 Summary 43

Chapter 4 Control Implementation and Simulation 44

4.1 Implementation 44
4.2 State Estimation 44
4.3 Speed Dependent Pitch Limit 45
4.4 Speed Compensated Trajectory Generator 46
4.5 Minimizing Actuator Motions 48
4.6 Robust Sliding Control Simulation 50
4.6.1 Speed Changes 50
4.6.2 Eliminating Noise Driven Actuator Motion 55
4.6.3 Adaptive Sliding Control 60
4.7 Summary 61

Chapter 5 Directional Stability 68

5.1 Model for Pitch 69
5.2 Model for Yaw 69
5.3 Hydrodynamic Stability 70
5.4 Dieudonne' Spiral Maneuver 74
5.5 Summary 82

Chapter 6 Conclusions and Future Work 83

6.1 Actuator Filtering 83
6.2 Directional Stability 84
6.3 Controller 85

Appendix 87

Bibliography 101

List of Figures and Tables

Figure 1.1 Odyssey Homing to the Dock....10

Figure 1.2 Odyssey Latched on the Dock....11

Figure 1.3 Odyssey Docked....12

Figure 2.1 Vehicle Coordinate System....15

Figure 4.1 Trapezoidal Trajectory Generator. 48

Figure 4.2 Comparing Performance With and Without Limited Elevator Motion. 50

Figure 4.3 Performance with Varying Speed. 52

Figure 4.4 Expanded View of Figure 4.3- transient at 1 m/sec. 53

Figure 4.5 Control Inputs for Mission in Figure 4.3. 54

Figure 4.6 Control Inputs for Transient in Figure 4.4- transient at 1 m/sec. 55

Figure 4.7 Comparing Performance With and Without Actuator Filter. 57

Figure 4.8 Expanded Portion of Actuator Filter Comparison - one transient at 1 m/sec. 58

Figure 4.9 Expanded Portion of Actuator Filter Comparison - beginning of transient. 59

Figure 4.10 Expanded Portion of Actuator Filter Comparison - end of transient. 60

Figure 4.11 Comparison of Performance With and Without Adaptation. 64

Figure 4.12 Adaptation of Heading Model Terms. 65

Figure 4.13 Adaptation of Pitch Model Terms. 66

Figure 4.14 Expanded Portion of Figure 4.11 - heading change with control inputs. ...67

Figure 4.15 Feedback for Heading Changes. 68

Table 5.1:Model Parameters 69

Figure 5.1 Hull Form with Fixed Fins and Latch. 74

Figure 5.2 Yaw Plane Pole Plot Showing Effect of Vertical Fins. 75

Figure 5.3 Typical Dieudonne' Spiral Maneuver Plot. 77

Figure 5.4 Simulated Dieudonne Plot. 78

Figure 5.5 Odyssey Dieudonne' Spiral Maneuver Plot. 79

Figure 5.6 Roll and Elevator Motion for Fixed Fin Evaluation. 80

Figure 5.7 Shift in Roll Angle at Zero Yaw Rate. 81

Figure 5.8 Comparison of Simulated Heading Performance with Vertical Fins. 82

Chapter 1

Thesis Description

A review of the Odyssey IIb autonomous underwater vehicle shows that energy costs associated with vehicle controls can be reduced and operational flexibility improved with relatively simple, low cost improvements. Because the operating speed that minimizes forward drag is not necessarily the same as that required for optimum sensor performance[2][19][20], a variable speed capability extending to the bottom of the vehicle speed range is sought. Optimizing Odyssey IIB AUV performance for slower speed operations and extended duration missions necessitates a multi-disciplinary review including control system design, hydrodynamic performance and sensor selection and utilization. Reducing the vehicle controls-fixed directional instability by adding vertical fixed fins, implementing an actuation filter, and designing a model based adaptive sliding controller improves the variable speed performance and reduces the control actuation necessary to provide the desired performance level with energy savings.

1.1 Motivation

A four dimensional ocean sampling project called the Autonomous Oceanographic Sampling Network (AOSN) is underway with MIT Sea Grant College Program, Woods Hole Oceanographic Institution and North Carolina State University as joint participants. The goal is an economically feasible system for repeated synoptic characterization of large scale oceanographic features[5]. A network consists of nodes with a varying number of base buoys and small autonomous underwater vehicles (AUVs). The AUVs sample paths in the network at desired intervals. Moored base buoys provide power and communi-

cation nodes to complete the long term multiple vehicle sampling network. Sampled parameter gradient error bounds determine vehicle population, tracklines, instruments, node distribution and sampling frequency. With the network in place multiple platforms for direct measurement exist within the defined ocean volume. This network's advantages are synoptic volume coverage, adaptive sampling, flexible control, energy management and robustness to component failure.

The Odyssey IIb vehicle, developed by the MIT Sea Grant Underwater Vehicles Laboratory, is planned for use in the AOSN. This AUV is a 2.15m long, 0.59m diameter cylindrical vehicle with a single thruster for propulsion and cruciform shaped control surfaces mounted in the stern section. The four control surfaces are vertically and horizontally mounted and are actuated in pairs.

The planned configuration for the first system deployment in the Labrador Sea is for two moored buoys to provide multiple services to two deployed vehicles. These services include an AUV docking station, navigation beacon, power source for recharging AUV batteries, and a data storage/transfer capability. Mooring buoys will be located within a Long Baseline Navigation network encompassing the area of interest. The AUVs will periodically undock, run a programed mission profile, collect oceanographic data for about four hours and then return to the dock. The AUV navigates to within a kilometer of a mooring buoy and then uses an Ultra Short Base Line navigation system to home to the dock. After docking, the AUV off-loads sampled data, recharges its batteries and then sleeps until the next scheduled sampling period. Figure 1.1 through Figure 1.3 illustrate the vehicle configuration and the docking sequence. In the first deployment, this routine is anticipated to continue for four months in the Labrador Sea through the winter and early

spring months. The vehicles will aid parameterizing the thermocline breakdown, convective plume formation, and their interaction with the meso-scale eddies in which they are embedded[13].

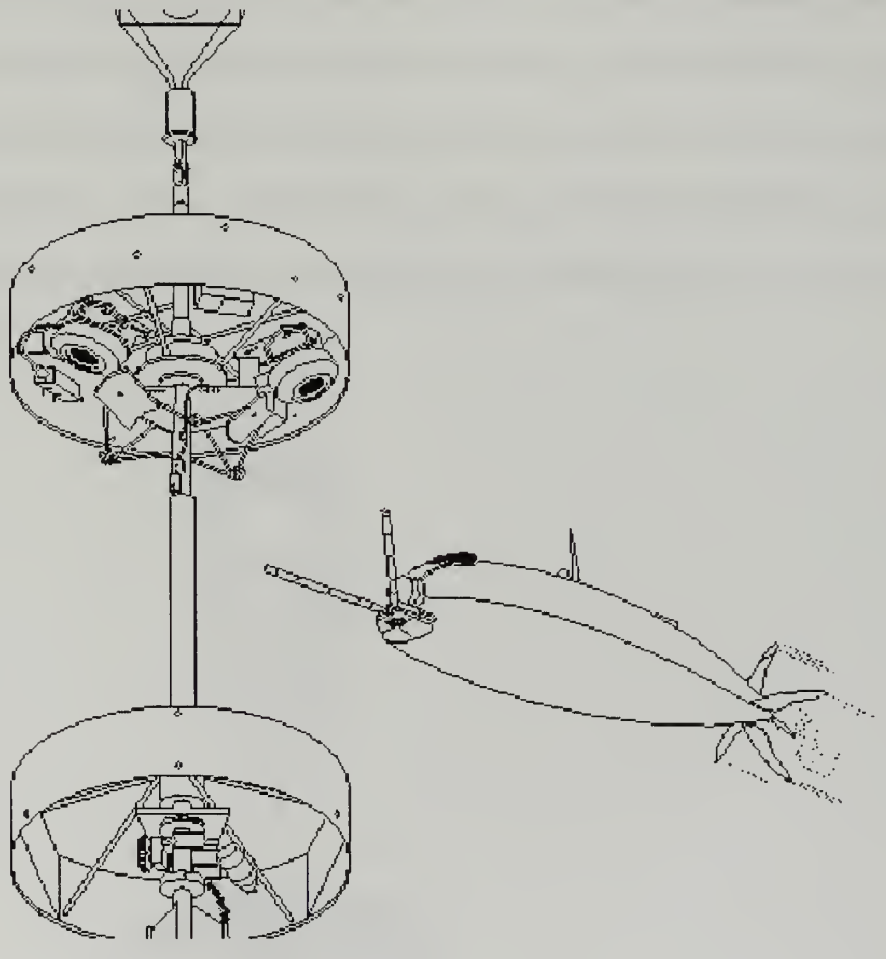

Figure 1.1 Odyssey Homing to the Dock

One vehicle specific problem encountered in Odyssey operations is the need for vehicle controller retuning. Five Odyssey IIb vehicles built by the MIT Sea Grant Lab, employed for various projects, require a different instrumentation set for most tasks. Each mission, with new instrumentation, different mounting configurations, different operating speeds, etc., requires controller retuning for optimal performance. Control gains are

unique to each vehicle and can not be applied to vehicles uniformly [1]. These complications limit operating speed choices to the one selected for optimum survey results.

Several vehicle behaviors can benefit from the ability to increase or decrease speed. During docking, slowing to reduce impact velocity will extend component lifetimes and increase the final homing maneuver effectiveness. Obstacle avoidance behaviors can increase speed to improve vehicle maneuverability. Bottom following algorithms can incorporate speed increases/decreases to compensate for changing needs for control authority[3].

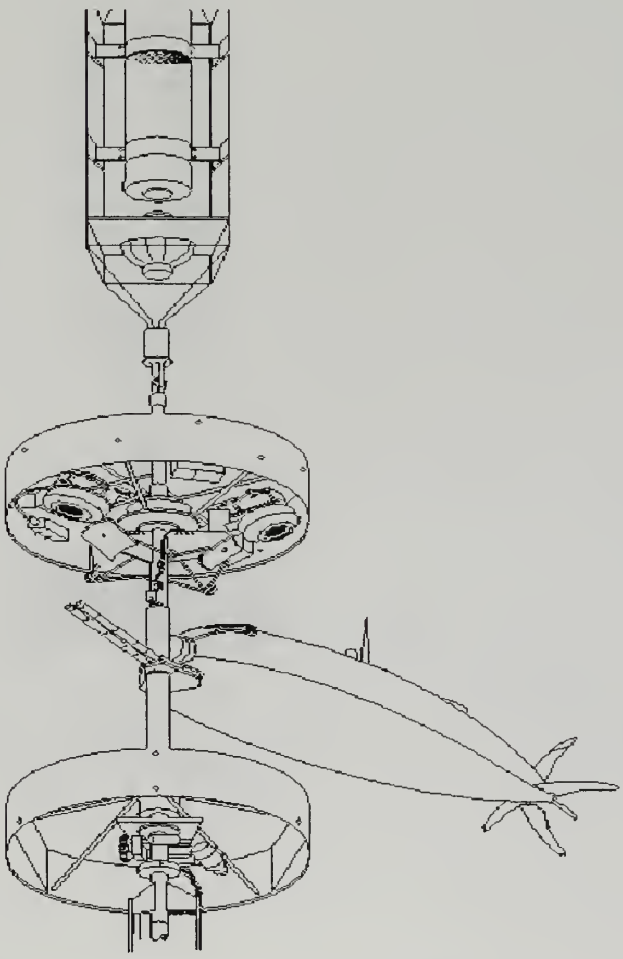

Figure 1.2 Odyssey Latched on the Dock

One concern for successful long term vehicle deployment for the AOSN project as well as long term AUV success is increased mission duration[16][29]. The present Silver-Zinc batteries in the Odyssey have insufficient battery capacity for the expected mission duration and required reserve capacity. For the Labrador Sea deployment, a larger, higher capacity Silver-Zinc battery will be used [18]. With limited payload capacity, increasing battery size is a limited option for solving the mission duration problem.

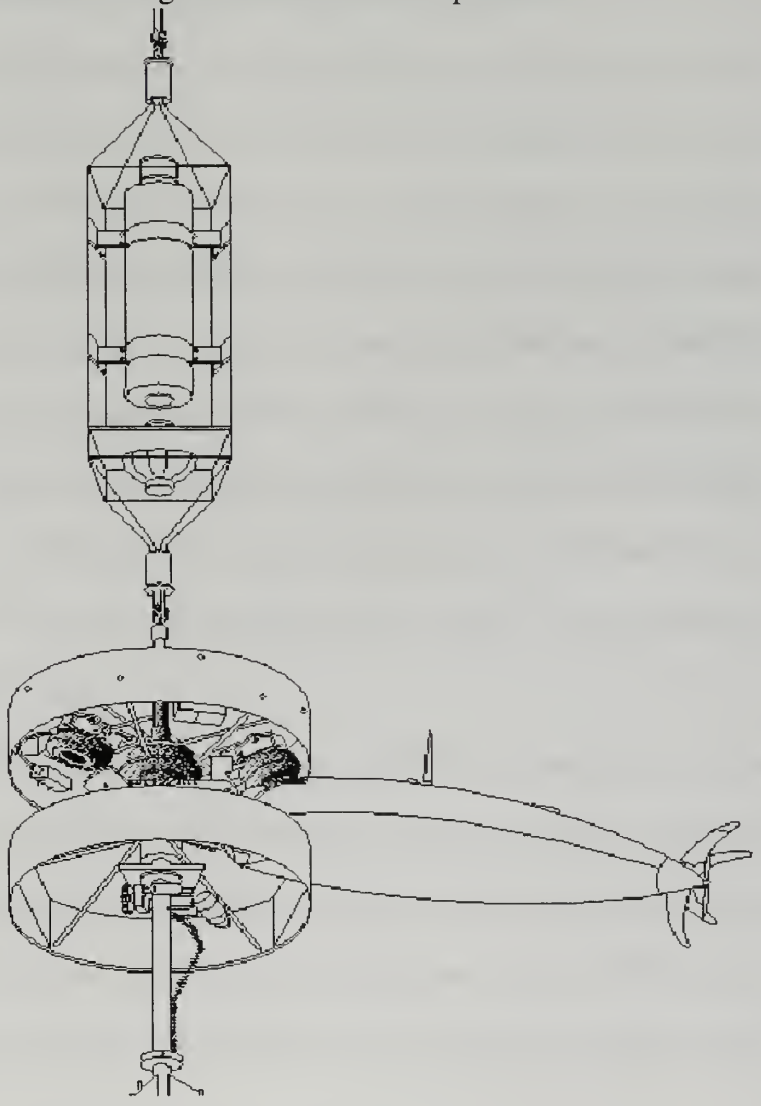

Figure 1.3 Odyssey Docked

1.2 Research Objectives

This thesis project has multiple goals of making vehicle improvements that enable changing scientific equipment configurations with minimal controller maintenance and allow vehicle speed flexibility during a mission to minimize power consumption while meeting sampling requirements. With the limitation that improvements will be relatively low cost and easy to implement, the concerns addressed by battery[17] and fuel cell[15][24] research are left alone. Long term, large scale changes that improve drag characteristics[12][16] are not in the scope of possible solutions for the same reason. What remains is the energy expended positioning the control surfaces. Although actuator energy requirements are a fraction of the propulsion load, for a small vehicle such as Odyssey they are significant and more efficient actuator use is beneficial. Any new controller design effort must provide robust performance for uncertain, changing hydrodynamic characteristics and varying operating speeds without controller retuning. Because it is doubtful that controller design alone will meet the comprehensive goals, the review includes hydrodynamic characteristics and instrumentation. Small improvements in several areas are expected to reinforce each other and the cumulative effect enable robust operations at varying speeds with energy savings.

With a large number of torpedo or tear drop shaped AUVs, the results of this effort apply to a large portion of the present AUV population. Some other AUVs sharing this basic shape include the AUSS (Naval Command, Control & Ocean Surveillance Center, San Diego, CA)[25], ORCA (Naval Research Lab, Stennis Space Center, MS)[4], Ocean Explorer (Florida Atlantic University)[22], REMUS (Woods Hole Oceanographic Institu-

tion)[26], the XP-21 vehicle (Applied Remote Technology), the UUV vehicles (Draper Laboratory) and the LDUUV (Naval Underwater Weapons Center, Newport RI)[28].

1.3 Thesis Outline

Chapter 2 discusses the standard submarine equations of motion and the model development used in controller design. Model simplifications from the full six degree of freedom standard submarine equations are presented.

Chapter 3 discusses the sliding controller design and the basics of that design approach for pitch/depth and yaw. After the sliding control design is developed, the adaptation addition is presented.

Additional implementation features and simulated vehicle performance with the sliding controller is offered in Chapter 4. Each feature is analyzed, isolating its effects, to assess its contribution.

Chapter 5 evaluates directional stability and reviews performance results after adding fixed vertical fins on the afterbody. It discusses the benefits obtained in the vertical and horizontal plane.

Conclusions and recommendations for modification implementation are summarized in Chapter 6.

Chapter 2

AUV Dynamics

2.1 Equations of Motion-Standard Submarine Equations

This chapter examines vehicle dynamics and adapts the Odyssey model developed by Hover [10]. First, the full six degree of freedom model is examined and then simplified. Equations are then uncoupled for independent models for pitch and yaw.

2.1.1 Coordinate System

The coordinate system is the standard submersible coordinate system and varies from that used in [6]. In this inverted coordinate system z-position increases with increasing depth. The right hand rule applies to this coordinate system as Figure 2.1 illustrates.

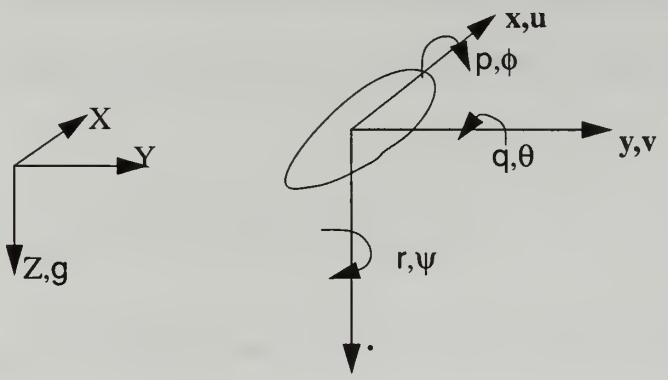

Figure 2.1 Vehicle Coordinate System

The three Euler angles, describe the coordinate transformation from an earth-fixed to body-fixed coordinate system. The yaw angle(Psi) is the angle about the z-axis, with positive angles the same as that read off a compass. Pitch (theta) is the second angle of rotation

and is measured about the transverse axis, y, so that positive pitch angles are attained with the vehicle pitched nose up. The final rotation, roll (Phi), is about the longitudinal axis with a roll to starboard being positive. These Euler angles, taken in the prescribed order uniquely describe all vehicle attitudes except for +/- 90 degree pitch angles. Odyssey can not achieve this condition so it is ignored.

Looking first at the forces acting on the vehicle, the translation and moment equations take the form shown in equations (2.1) and (2.2). This includes the vehicle body, $\vec{F}_{D,L,Am}$, weight, $\vec{F}_W$, buoyancy, $\vec{F}_B$, propulsion, $\vec{F}t$ and control surface effects, $\vec{D}f\vec{L}f$.

$$-\vec{F}i = \vec{F}_{D,L,Am} + \vec{F}_W + \vec{F}_B + \vec{F}t + \vec{D}f + \vec{L}f \tag{2.1}$$

$$-\vec{M}i = \vec{M}_{D,L,Am} + \vec{M}_W + \vec{M}_B + \vec{M}t + \vec{r}f \times (\vec{D}f + \vec{L}f) \tag{2.2}$$

With the vector $\vec{r}_{cm}$ representing the center of mass position measured from the origin of the body-reference system, the acceleration vector at the mass center in body-referenced coordinates is:

$$\vec{a}_{cm} = \dot{\vec{v}} + \vec{w} \times \vec{v} + \dot{\vec{w}} \times \vec{r}_{cm} + \vec{w} \times (\vec{w} \times \vec{r}_{cm}) \tag{2.3}$$

The vector $\vec{r} = \begin{bmatrix} x & y & z \end{bmatrix}$ is the vehicle position in the earth coordinate system. Eq (2.4) represents the vehicle's body referenced velocities. The vehicle angular rotation vector through the Euler angles is represented by $\vec{\omega}$ shown in Eq. (2.5).

$$\vec{v} = \begin{bmatrix} u & v & w \end{bmatrix} \tag{2.4}$$

$$\vec{\omega} = \begin{bmatrix} p & q & r \end{bmatrix} \tag{2.5}$$

When working with a symmetrical body, it is convenient to have the origin be the longitudinal, transverse and normal body axes intersection. In this situation, if the cross moments of inertia are negligibly small then the rigid body equations of motion are expressed as:

$$\vec{F}i = -m\begin{bmatrix} 1 & 0 & 0 & 0 & z_{cm} & -y_{cm} \\ 0 & 1 & 0 & -z_{cm} & 0 & x_{cm} \\ 0 & 0 & 1 & y_{cm} & -x_{cm} & 0 \end{bmatrix}\begin{bmatrix} \dot{\vec{v}} \\ \dot{\vec{w}} \end{bmatrix} - m\begin{bmatrix} pqy_{cm} - q^2x_{cm} - r^2x_{cm} + prz_{cm} + qw - rv \\ qrz_{cm} - r^2y_{cm} - p^2y_{cm} + pqx_{cm} + ru - pw \\ prx_{cm} - p^2z_{cm} - q^2z_{cm} + qry_{cm} + pv - qu \end{bmatrix} = \begin{bmatrix} \vec{X} \\ \vec{Y} \\ \vec{Z} \end{bmatrix} \tag{2.6}$$

$$\vec{M}i = -I\vec{\omega} - \vec{\omega} \times (I\vec{\omega}) - m\vec{r}_{cm} \times (\dot{\vec{v}} + \overrightarrow{\vec{\omega} \times \vec{v}}) \tag{2.7}$$

$$\vec{M}i = -I\vec{\omega} - \begin{bmatrix} 0 & -Izx & Ixy & -Iyz & Izz-Iyy & Iyz \\ Izx & Iyz & Ixx-Izz & 0 & -Ixy & -Ixz \\ -Ixy & Iyy-Ixx & -Iyz & Ixy & Ixz & 0 \end{bmatrix}\begin{bmatrix} p^2 \\ pq \\ pr \\ q^2 \\ qr \\ r^2 \end{bmatrix}$$

$$-m\begin{bmatrix} 0 & -z_{cm} & y_{cm} \\ z_{cm} & 0 & -x_{cm} \\ -y_{cm} & x_{cm} & 0 \end{bmatrix}\dot{\vec{v}} - m\begin{bmatrix} 0 & pv-qu & pw-ru \\ qu-pv & 0 & qw-rv \\ ru-pw & rv-qw & 0 \end{bmatrix}\vec{r}_{cm} = \begin{bmatrix} \vec{K} \\ \vec{M} \\ \vec{N} \end{bmatrix} \tag{2.8}$$

2.1.2 Coordinate Transformation

The transformation matrix converting the vehicles trajectory from an earth-fixed reference frame to a body-fixed reference frame is given in Eq. (2.9).

$$\vec{x}_{body} = \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\cos\theta \\ \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\cos\theta \end{bmatrix}\vec{x}_{inertial} \tag{2.9}$$

Where C is the matrix defined in Eq. (2.9),

$$\vec{x}_{body} = C \cdot \vec{x}_{inertial} \tag{2.10}$$

The corresponding transformation matrix for the Euler angle rates in transformation matrix terms is similar.

$$\dot{\vec{E}} = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi/\cos\theta & \cos\phi/\cos\theta \end{bmatrix} \vec{\omega} = \Gamma(\vec{E})\vec{\omega} \tag{2.11}$$

2.1.3 Weight and Buoyancy

Weight and Buoyancy forces in earth coordinates are written here.

$$\vec{F}_W = C \begin{bmatrix} 0 \\ 0 \\ W \end{bmatrix} \tag{2.12}$$

$$\vec{F}_B = C \begin{bmatrix} 0 \\ 0 \\ -B \end{bmatrix} \tag{2.13}$$

Moments are obtained with the vector cross products of $\vec{r}_W$ and $\vec{r}_B$.

$$\vec{M}_W = \begin{bmatrix} y_W F_{Wz} - z_W F_{Wy} \\ z_W F_{Wx} - x_W F_{Wz} \\ x_W F_{Wy} - y_W F_{Wx} \end{bmatrix} \tag{2.14}$$

$$\vec{M}_B = \begin{bmatrix} -y_B F_{Bz} + z_B F_{By} \\ -z_B F_{Bx} + x_B F_{Bz} \\ -x_B F_{By} + y_B F_{Bx} \end{bmatrix} \tag{2.15}$$

2.2 Linearized Dynamics

At this point the rigid body dynamics and the most straightforward forces have been considered. For the remaining forces a theoretical and experimental method combination is used. Hover uses a variety of methods [10]to model fluid forces. Hoerner's curves [8][9]are used to develop lift and drag expressions for the body. Newman's strip theory [14] is used for lateral drag and body and fin added mass calculations. Momentum theory [27] is used for propulsion forces. Common models for fin forces are taken from [7].

Dynamics are linearized because linear calculations give insight into basic dynamic response. This method shows directional stability and non-minimum phase behavior. Hover also opined [10] that when in straight and level flight the multi-input, multi-output linear control methodology works well. Model linearization about an operating speed is not limiting in practice. Operationally, speed is a fairly constant parameter and changes, typically made in a step-wise manner, are relatively infrequent. Gain scheduling using an array of operating speeds produces a gain structure that gives the desired performance. This unpopular scheme is complex and requires repetitive design effort at each operating point. The guaranteed tracking performance with modeling errors and parameter uncertainty makes the sliding controller an attractive option. With drag coefficients being Reynolds number functions, each new operating speed introduces new parameter uncertainties. Adding the adaptation scheme to the sliding controller furthers the capability to provide guaranteed performance as the parameter uncertainty due to changing speeds or modeling errors is transparent to the controller.

The dynamics are linearized with yaw and pitch coupled because axial torque from the thruster leads to a static roll angle which cannot be corrected when elevator and rudder

surfaces are ganged. Cross coupling between elevator motion and heading and rudder motion and pitch is assumed to be small. In practice, a primary vehicle ballasting consideration is minimizing cross coupling. The decoupled equations are found by setting the static roll angle ϕ to zero. Also the variables are linearized about an equilibrium point by using an overbar ($\bar{\theta}$) to represent equilibrium and a tilde ($\tilde{\theta}$) to represent deviations from equilibrium. Assumptions made for linearizing are:

1. $|\vec{v}| \cong |\dot{\vec{v}}| \cong \bar{u}$

2. Pitch $\tilde{\theta}$ and yaw $\tilde{\psi}$ deviations are small.

3. Second and higher order terms are negligibly small.

With these assumptions the Inertial Forces and Moments linearization in steady flight becomes:

$$\vec{F}i = -m\begin{bmatrix} 1 & 0 & 0 & 0 & z_{cm} & -y_{cm} \\ 0 & 1 & 0 & -z_{cm} & 0 & x_{cm} \\ 0 & 0 & 1 & y_{cm} & -x_{cm} & 0 \end{bmatrix}\begin{bmatrix} \dot{\tilde{v}} \\ \dot{\tilde{w}} \end{bmatrix} - m\begin{bmatrix} 0 \\ r\bar{u} \\ -q\bar{u} \end{bmatrix} \tag{2.16}$$

The rotational terms are:

$$\vec{M} = -I\dot{\vec{w}} - m\begin{bmatrix} 0 & -z_{cm} & y_{cm} \\ z_{cm} & 0 & -x_{cm} \\ -y_{cm} & x_{cm} & 0 \end{bmatrix}\dot{\tilde{v}} - m\begin{bmatrix} 0 & -q\bar{u} & -r\bar{u} \\ q\bar{u} & 0 & 0 \\ r\bar{u} & 0 & 0 \end{bmatrix}\vec{r}_{cm} \tag{2.17}$$

Yaw and Pitch angles are assumed to be small, while the static roll angle may be ten degrees or more, usually due to induced torque from the thruster. For small pitch and yaw perturbations, the coordinate transformation matrix can be written as the sum of $\bar{C}$ and $\tilde{C}$.

$$\bar{C} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\bar{\phi} & \sin\bar{\phi} \\ 0 & -\sin\bar{\phi} & \cos\bar{\phi} \end{bmatrix} \tag{2.18}$$

$$\tilde{C} = \begin{bmatrix} 0 & \psi & -\theta \\ \sin\bar{\phi}\theta - \cos\bar{\phi}\psi & 0 & 0 \\ \cos\bar{\phi}\theta + \sin\bar{\phi}\psi & 0 & 0 \end{bmatrix} \tag{2.19}$$

2.3 Weight and Buoyancy

Using the transformation matrix decomposed form, weight and buoyancy terms are expressed as

$$\overrightarrow{Fw} = (\bar{C} + \tilde{C}) \begin{bmatrix} 0 \\ 0 \\ W \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ W \end{bmatrix} + \begin{bmatrix} -W\theta \\ 0 \\ 0 \end{bmatrix} \tag{2.20}$$

$$\overrightarrow{F_B} = (\bar{C} + \tilde{C}) \begin{bmatrix} 0 \\ 0 \\ -B \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -B \end{bmatrix} + \begin{bmatrix} B\theta \\ 0 \\ 0 \end{bmatrix} \tag{2.21}$$

$$\overrightarrow{Fw} + \overrightarrow{F_B} = \begin{bmatrix} 0 \\ 0 \\ W-B \end{bmatrix} + \begin{bmatrix} (B-W)\theta \\ 0 \\ 0 \end{bmatrix} \tag{2.22}$$

$$\vec{M}w = \begin{bmatrix} y_w\vec{F}_{wz} - z_w\vec{F}_{wy} \\ z_w\vec{F}_{wx} - x_w\vec{F}_{wz} \\ x_w\vec{F}_{wy} - y_w\vec{F}_{wx} \end{bmatrix} = \begin{bmatrix} y_G W \\ -x_G W \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ -z_G W\theta \\ y_G W\theta \end{bmatrix} \tag{2.23}$$

$$\vec{M}_B = \begin{bmatrix} -y_B\vec{F}_{Bz} + z_B\vec{F}_{By} \\ -z_B\vec{F}_{Bx} + x_B\vec{F}_{Bz} \\ -x_B\vec{F}_{By} + y_B\vec{F}_{Bx} \end{bmatrix} = \begin{bmatrix} -y_B B \\ x_B B \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ z_B B\theta \\ -y_B B\theta \end{bmatrix} \tag{2.24}$$

$$\vec{M}_W + \vec{M}_B = \begin{bmatrix} y_W W - y_B B \\ x_B B - x_W W \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ (z_B B - z_W W)\theta \\ (y_W W - y_B B)\theta \end{bmatrix} \tag{2.25}$$

2.4 Body and Fin Fluid Forces and Moments

The linearized expressions taken from [10] for the fluid forces and moments acting on the vehicle body are:

$$\tilde{\vec{F}}_{D,L,Am} = \begin{bmatrix} 2X_{u|u|}\bar{u}\tilde{u} + X_{\dot{u}}\dot{u} \\ Y_{uv}\bar{u}v + Y_{\dot{v}}\dot{v} + Y_{\dot{r}}\dot{r} \\ Z_{uw}\bar{u}w + Z_{\dot{w}}\dot{w} + Z_{\dot{q}}\dot{q} \end{bmatrix} \tag{2.26}$$

$$\tilde{\vec{M}}_{D,L,Am} = \begin{bmatrix} K_{\dot{p}} \\ 2M_{u|u|}\bar{u}\tilde{u} + M_{uw}\bar{u}w + M_{\dot{w}}\dot{w} + M_{\dot{q}}\dot{q} \\ 2N_{u|u|}\bar{u}\tilde{u} + N_{uv}\bar{u}v + N_{\dot{v}}\dot{v} + N_{\dot{r}}\dot{r} \end{bmatrix} \tag{2.27}$$

In a linear approximation during steady flight, fin forces consist only of lift and are written as follows using the subscript notation X_{ab}. The a is either r for rudder or e for elevator. The b is a number one or two to differentiate the two fins.

- Z_{e1} Z_{e2} Y_{r1} Y_{r2} Forces from the two fins
- ρ fluid density
- A_{e1} A_{e2} A_{r1} A_{r2} Fin Surface Areas
- Cl_{e1} Cl_{e2} Cl_{r1} Cl_{r2} Lift coefficients
- δ_{e1} δ_{e2} δ_{r1} δ_{r2} Fin Position in radians

$$Z_{e1} = -0.5\rho A_{e1} Cl_{e1}(\bar{u}w + \bar{u}y_{e1}p - \bar{u}x_{e1}q + \bar{u}^2\delta_{e1}) \tag{2.28}$$

$$Z_{e2} = -0.5\rho A_{e2} Cl_{e2}(\bar{u}w + \bar{u}y_{e2}p - \bar{u}x_{e2}q + \bar{u}^2\delta_{e2}) \tag{2.29}$$

$$Y_{r1} = 0.5\rho A_{r1} Cl_{r1}(-\bar{u}v - \bar{u}w_{r1}r + \bar{u}z_{r1}p) + \bar{u}^2\delta_{r1} \tag{2.30}$$

$$Y_{r2} = 0.5\rho A_{r2} Cl_{r2}(-\bar{u}v - \bar{u}w_{r2}r + \bar{u}z_{r2}p) + \bar{u}^2\delta_{r2} \tag{2.31}$$

The moments for the rudder and elevator are applied on the appropriate moment arms.

2.5 Linearized Model for Control System Design

Hover considers only a neutrally buoyant vehicle in his model development so Weight and Buoyancy terms are absent in his model. The usual practice is to have the vehicle ballasted slightly positive for easier recovery. This practice significantly affects vehicle dynamics when slower operating speeds are attempted. With positive buoyancy considered, the linearized model becomes:

$$I\dot{\vec{x}} = A\vec{x} + B\vec{u} \tag{2.32}$$

$$\vec{x} = \begin{bmatrix} u & v & w & p & q & r & x & y & z & \phi & \theta & \psi \end{bmatrix}^T \tag{2.33}$$

$$\vec{u} = \begin{bmatrix} \delta r & \delta e & 1 \end{bmatrix}^T \tag{2.34}$$

(2.35)

$$
A = \begin{bmatrix}
2X_{uu}U & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & Y_{uv}U - C_{fin}U & 0 & 0 & 0 & Y_{ur}U - mU - C_{fin}U_{x_{fin}} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & Z_{uw}U - C_{fin}U & 0 & -Z_{uq}U + mU + C_{fin}U_{x_{fin}} & 0 & 0 & 0 & 0 & B-W & 0 \\
0 & 0 & 0 & 0 & mU_{y_G} & mU_{z_G} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -M_{uw}U + C_{fin}U_{x_{fin}} & 0 & M_{uq}U - mU_{x_G} - C_{fin}U_{x_{fin}}{}^2 & 0 & 0 & 0 & 0 & z_B B - z_w W & 0 \\
0 & N_{uv}U - C_{fin}U_{x_{fin}} & 0 & 0 & 0 & N_{ur}U - mU_{x_G} - C_{fin}U_{x_{fin}}{}^2 & 0 & 0 & 0 & y_w W - y_B B & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \cos\phi & -\sin\phi & 0 & 0 & 0 & 0 & 0 & 0 & 0 & U \\
0 & \sin\phi & \cos\phi & 0 & 0 & 0 & 0 & 0 & 0 & -U & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \cos\phi & -\sin\phi & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \sin\phi & \cos\phi & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

$$
I = \begin{bmatrix}
m + Mau & 0 & 0 & 0 & 0 & mz_G & -my_G & 0\,0\,0\,0\,0 \\
0 & m + Mav & 0 & -mz_G & 0 & mx_G + xmaMav & 0\,0\,0\,0\,0 \\
0 & 0 & m + Maw & my_G & -mx_G - xmaMav & 0 & 0\,0\,0\,0\,0 \\
0 & -mz_G & my_G & J_{xx} + Jamx & 0 & 0 & 0\,0\,0\,0\,0 \\
mz_G & 0 & -mx_G - xmaMav & 0 & J_{yy} + Jamy & 0 & 0\,0\,0\,0\,0 \\
-my_G & mx_G + xmaMav & 0 & 0 & 0 & J_{zz} + Jamz & 0\,0\,0\,0\,0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1\,0\,0\,0\,0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0\,1\,0\,0\,0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0\,0\,1\,0\,0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0\,0\,0\,1\,0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0\,0\,0\,0\,1
\end{bmatrix}
\tag{2.36}
$$

(2.37)

$$B = \begin{bmatrix} -0.5\rho A_{fin} C d_{fin} 2U & -0.5\rho A_{fin} C d_{fin} 2U & 0 \\ C_{fin} U^2 & 0 & 0 \\ 0 & -C_{fin} U^2 & W-B \\ 0 & 0 & y_W W - y_B B \\ 0 & x_{fin} C_{fin} U^2 & x_B B - x_W W \\ x_{fin} C_{fin} U^2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

The buoyancy and weight terms are split between the A and B matrices for convenience. The vertical moment arm is included in the A matrix and is one of the terms adapted for control purposes. The other buoyancy and weight term is more accurately determined prior to vehicle deployment. It is considered a constant pitch input and is included in the B matrix.

2.6 Models for Pitch and Yaw

Separate uncoupled models for pitch and yaw are presented here. With the vertical and horizontal control surfaces ganged, pitch is uncontrollable. This makes uncoupled models an attractive model simplification. Maintaining a separation between the center of gravity and buoyancy effectively limits the roll angles to less than ten degrees. The vehicle states chosen are those necessary to show the vehicle instability in the yaw plane in the linearized model. The corresponding pitch plane parameters are used for a similar pitch plane model.

2.6.1 Model for Pitch

$$I_\theta \dot{\hat{x}} = A_\theta \hat{x} + B_\theta \hat{u} \tag{2.38}$$

$$\hat{x} = \begin{bmatrix} w & q & \theta \end{bmatrix}^T \tag{2.39}$$

$$u = \begin{bmatrix} \delta e & 1 \end{bmatrix}^T \tag{2.40}$$

$$I_\theta = \begin{bmatrix} m + Maw & -mx_G - x_{ma}Maw & 0 \\ -mx_G - x_{ma}Maw & Jyy + Jamy & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.41}$$

$$A_\theta = \begin{bmatrix} ZuwU - C_{fin}U & -ZuqU + mU + C_{fin}Ux_{fin} & B-W \\ -MuwU + C_{fin}Ux_{fin} & MuqU - mUx_G - C_{fin}Ux_{fin}^2 & z_B B - z_G W \\ 0 & \cos\phi & 0 \end{bmatrix} \tag{2.42}$$

$$B_\theta = \begin{bmatrix} -C_{fin}U^2 & W-B \\ x_{fin}C_{fin}U^2 & x_B B - x_w W \\ 0 & 0 \end{bmatrix} \tag{2.43}$$

2.6.2 Model for Yaw

$$I_\psi \dot{\hat{x}} = A_\psi \hat{x} + B_\psi \hat{u} \tag{2.44}$$

$$\hat{x} = \begin{bmatrix} v & r & \psi \end{bmatrix}^T \tag{2.45}$$

$$u = \delta r \tag{2.46}$$

$$I_\psi = \begin{bmatrix} m + Mav & mx_G + x_{ma}Mav & 0 \\ mx_G + x_{ma}Mav & Jzz + Jamz & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.47}$$

$$A_\psi = \begin{bmatrix} YuvU - C_{fin}U & YurU - mU - C_{fin}Ux_{fin} & 0 \\ NuvU - C_{fin}Ux_{fin} & NurU - mUx_G - C_{fin}Ux_{fin}^2 & 0 \\ 0 & \cos\phi & 0 \end{bmatrix} \tag{2.48}$$

$$B_\psi = \begin{bmatrix} C_{fin}U^2 \\ x_{fin}C_{fin}U^2 \\ 0 \end{bmatrix}$$

2.7 Summary

The models for pitch and yaw are the linearized models used by Hover for the PID controller design with weight and buoyancy added. The hydrodynamic parameter estimates using Hoerner's curves, strip theory and the like, were refined in the tow tank at Draper Lab using a 1.45 m/sec forward speed[11]. These values are used later to show the yaw plane instability. With two uncoupled models for pitch and yaw that take into account net positive buoyancy and vehicle dynamics in sufficient detail, the controller design task presents itself.

Chapter 3

Control System Design

Sliding mode control is a nonlinear tracking control system method. Given a nonlinear model with bounded parameter uncertainty and bounded disturbances, the method yields a nonlinear feedback controller with guaranteed tracking and stability properties. The system is designed to follow a commanded state trajectory and drive the tracking error to zero. The tracking error dynamics are actually specified and used as a constraint on the system. The object is, given the given tracking error dynamics, define a control law such that those dynamics are achieved by the closed loop system with modeling errors and disturbances. The sliding controller development presented here is adapted from Slotine and Li [21].

Sliding control is formulated in this chapter for heading and depth/pitch control. Speed control is not addressed as the present instrumentation is inadequate for a constant speed controller. The vehicle uses open loop speed control because speed is not measured in any axis.

3.1 Robust Sliding Control Overview

Consider a single input dynamic system

$$x^n = f(x) + b(x)u$$

where x is the scalar of interest, the scalar u is the control input and the state vector $\mathbf{x_d} = \left[x_d \; \dot{x}_d \; \ddot{x}_d \; \ldots \; x_d^{(n-1)}\right]^T$ results. The terms f and b are generally nonlinear state vector

functions and need not be exactly known. What is known about f and b is that they are upper bounded by a known continuous function of x. Additionally b is of known sign. The control problem is to get the state vector x to track a specific time varying state x_d in the presence of model imprecision on $f(x)$ and $b(x)$.

Looking at notation, let $\tilde{x} = x - x_d$ be the tracking error in the variable x and $\tilde{\mathbf{x}} = \left[\tilde{x}\ \dot{\tilde{x}}\ \ddot{\tilde{x}}\ \ldots\ \tilde{x}^{(n-1)}\right]^T$ be the tracking error vector. Defining a time varying surface $S(t)$ in the state space R^n by the scalar equation $s(x;t) = 0$ where the desired error dynamics are defined to be of the form

$$s(x;t) = \left(\frac{d}{dt} + \lambda\right)^{n-1} \tilde{x} \tag{3.1}$$

The term λ is a strictly positive constant and is considered the control bandwidth as an analogy to linear control theory. It is the weighting factor in the weighted sum of the position and velocity error. If $n = 2$, then $s = \dot{\tilde{x}} + \lambda\tilde{x}$. This corresponds to a line that moves with $\mathbf{x_d} = \left[x_d\ \dot{x}_d\right]^T$ and has a $-\lambda$ slope. The system state does not follow $\mathbf{x_d} = \left[x_d\ \dot{x}_d\right]^T$ exactly but remains on the line $s(x;t) = 0$. The desired closed loop dynamics $s = \dot{\tilde{x}} + \lambda\tilde{x} = 0$ are exactly achieved and accordingly $\tilde{x}(t) = \tilde{x}(0)e^{-\lambda t}$. The simplified first order problem of keeping $s = 0$ can now be achieved by choosing the control law u such that outside $S(t)$

$$\frac{1}{2}\frac{ds^2}{dt} \leq -\eta|s| \tag{3.2}$$

where η is a strictly positive constant. This states that the squared distance to the surface as measured by s^2 decreases along all system trajectories. Once on the surface, the system trajectories remain on the surface. Some disturbances or uncertainties can be tolerated,

while keeping the system trajectories on the surface. Eq. (3.2) is derived in [23] from Lyapunov theory and the discontinuous differential equation dynamics.

Thinking about the SISO system, the control gain $b(x)$ is not precisely known but is bounded, so that $0 < b_{min} \leq b \leq b_{max}$. Because the control input enters multiplicatively in the dynamics, our estimate $\hat{b}$ of gain b is the geometric mean of the bounds;

$$\hat{b} = (b_{min} b_{max})^{1/2} \tag{3.3}$$

Bounds can be written in the form,

$$\beta^{-1} \leq \frac{\hat{b}}{b} \leq \beta \tag{3.4}$$

where

$$\beta = (b_{max}/b_{min})^{1/2} \tag{3.5}$$

The term β is called the design gain margin as an analogy to linear control design.

Considering a second order system, $\ddot{x} = f + bu$. In order to have the system track $x = x_d$ the sliding surface is defined as $s = \dot{\tilde{x}} + \lambda\tilde{x}$. Then, we can say that

$$\dot{s} = \ddot{x} - \ddot{x}_d + \lambda\dot{\tilde{x}} = f + bu - \ddot{x}_d + \lambda\dot{\tilde{x}}. \tag{3.6}$$

The best continuous control law approximation that achieves $\dot{s} = 0$ is $\hat{u} = -\hat{f} + \ddot{x}_d - \lambda\dot{\tilde{x}}$. Adding a feedback term to satisfy sliding conditions because of uncertainties in f, the control law becomes $u = \hat{b}^{-1}[\hat{u} - k\,\mathrm{sgn}(s)]$.

Choosing the control law as stated leads to a discontinuous control action called dithering or chatter when the $s = 0$ state is reached. To avoid exciting unmodeled dynam-

ics a continuous control law is developed by establishing a boundary layer surrounding the switching surface. The layer is defined as

$$B(t) = \{x, |s(x;t)| < \Phi\} \tag{3.7}$$

where Φ is the boundary layer thickness. The control law becomes

$$u = \hat{b}^{-1}[\hat{u} - ksat(s/\Phi)] \tag{3.8}$$

The control discontinuity smoothing inside $B(t)$ essentially assigns a low pass filter structure to the s local dynamics, thus eliminating chatter.

3.2 Controller Design

In designing the model based adaptive sliding controller, first the robust sliding controller development is presented and then subsequently the adaptation method. The control model is further simplified by taking only the moment equations for both pitch and yaw. This is very straightforward for heading control. For depth and pitch control an early PID controller uses a transfer function to generate a desired pitch based on a depth error. The present PID controller uses an inner/outer loop control method for pitch and depth. This same approach is used for the sliding controller with some modifications.

3.2.1 Robust Sliding Controller Design for Pitch

Taking the moment equation for pitch, we can derive the robust sliding controller as follows:

$$\begin{bmatrix} -mx_G - x_{ma}Maw & Jyy + Jamy & 0 \end{bmatrix} \begin{bmatrix} \dot{w} \\ \dot{q} \\ \dot{\theta} \end{bmatrix} \tag{3.9}$$

$$= \begin{bmatrix} -MuwU + C_{fin}Ux_{fin} & MuqU - mUx_G - C_{fin}Ux_{fin}^2 & z_BB - z_GW \end{bmatrix} \begin{bmatrix} w \\ q \\ \theta \end{bmatrix} + \begin{bmatrix} x_{fin}C_{fin}U^2 & x_BB - x_WW \end{bmatrix} \begin{bmatrix} \delta e \\ 1 \end{bmatrix}$$

$$(-mx_G - x_{ma}Maw)\dot{w} + (Jyy + Jamy)\dot{q} = (-MuwU + C_{fin}Ux_{fin})w + (MuqU - mUx_G - C_{fin}Ux_{fin}^2)q$$

$$+ (z_BB - z_GW)\theta + (x_{fin}C_{fin}U^2)\delta e + x_BB - x_WW \tag{3.10}$$

Arranging the state variables in a linear combination

$$u = U \tag{3.11}$$

$$a1 = Jyy + Jamy \tag{3.12}$$

$$a2 = -Muq + mx_G + C_{fin}x_{fin}^2 \tag{3.13}$$

$$a3 = -mx_G - x_{ma}Maw \tag{3.14}$$

$$a4 = Muw - C_{fin}x_{fin} \tag{3.15}$$

$$a5 = z_BB - z_GW \tag{3.16}$$

$$a6 = (x_WW - x_BB) \tag{3.17}$$

$$a1\ddot{\theta} + a2u\dot{\theta} + a3\dot{w} + a4uw + a5\sin\theta + a6\cos\theta = -x_{fin}C_{fin}u^2\delta e = \Delta e \tag{3.18}$$

Defining the following terms:

$$\tilde{\theta} = \theta - \theta_d \tag{3.19}$$

$$\dot{\tilde{\theta}} = \dot{\theta} - \dot{\theta}_d \tag{3.20}$$

$$\ddot{\tilde{\theta}} = \ddot{\theta} - \ddot{\theta}_d \tag{3.21}$$

Here a reference velocity, $\dot{\theta}_r$, is formed by shifting the desired velocity, $\dot{\theta}_d$, according to the position error, $\tilde{\theta}$.

$$\dot{\theta}_r = \dot{\theta}_d - \lambda_\theta \tilde{\theta} \tag{3.22}$$

$$s_\theta = \dot{\tilde{\theta}} + \lambda_\theta \tilde{\theta} = \dot{\theta} - \dot{\theta}_r \tag{3.23}$$

$$\ddot{\theta}_r = \ddot{\theta}_d - \lambda_\theta \dot{\tilde{\theta}} \tag{3.24}$$

$$\dot{s_\theta} = \ddot{\theta} - \ddot{\theta}_r \tag{3.25}$$

Define a Lyapunov function candidate, V(t), a positive definite function:

$$V = \frac{1}{2} a1 (s_\theta)^2 \tag{3.26}$$

Differentiating,

$$\dot{V} = s_\theta a1 \dot{s_\theta} + \frac{1}{2} s_\theta \dot{a1}(s_\theta) \tag{3.27}$$

Substituting $\dot{s_\theta} = \ddot{\theta} - \ddot{\theta}_r$,

$$\dot{V} = s_\theta (a1\ddot{\theta} - a\dot{1}\dot{\theta}_r) + \frac{1}{2} s_\theta \dot{a1} s_\theta \tag{3.28}$$

Substituting $a1\ddot{\theta}$ from the system dynamics yields

$$a1\ddot{\theta} = \Delta e - (a2u\dot{\theta} + a3\dot{w} + a4uw + a5\sin\theta + a6\cos\theta) \tag{3.29}$$

Using the relations

$$s_\theta = \dot{\tilde{\theta}} + \lambda_\theta \tilde{\theta} = \dot{\theta} - \dot{\theta}_r \tag{3.30}$$

$$\dot{\theta} = s_\theta + \dot{\theta}_r \tag{3.31}$$

then

$$a1\ddot{\theta} = \Delta e - [(a2u(s_\theta + \dot{\theta}_r) + a3\dot{w} + a4uw + a5\sin\theta + a6\cos\theta)] \tag{3.32}$$

$$\dot{V} = s_\theta (\Delta e - [a2u(s_\theta + \dot{\theta}_r) + a3\dot{w} + a4uw + a5\sin\theta + a6\cos\theta] - a1\ddot{\theta}_r) + \frac{1}{2} s_\theta \dot{a1}(s_\theta) \tag{3.33}$$

$$\dot{V} = s_\theta (\Delta e - a1\ddot{\theta}_r - a2u\dot{\theta}_r - a3\dot{w} - a4uw - a5\sin\theta - a6\cos\theta) \tag{3.34}$$

Define the control input to be:

$$\Delta e = \hat{\Delta e} - k_\theta \operatorname{sgn}(s)) \tag{3.35}$$

If the dynamics are exactly known, $\hat{\Delta e}$ is the control input that makes $\dot{V}$=0.

$$\hat{\Delta e} = \hat{a1}\ddot{\theta}_r + \hat{a2}u\dot{\theta}_r + \hat{a3}\dot{w} + \hat{a4}uw + \hat{a5}\sin\theta + \hat{a6}\cos\theta \tag{3.36}$$

We then have,

$$\dot{V} = s_\theta(\hat{\Delta}e - k_\theta \operatorname{sgn}(s)) - a1\ddot{\theta}_r - a2u\dot{\theta}_r - a3\dot{w} - a4uw - a5\sin\theta - a6\cos\theta) \tag{3.37}$$

$$\dot{V} = s_\theta(\Delta e - \tilde{a1}\ddot{\theta}_r - \tilde{a2}u\dot{\theta}_r - \tilde{a3}\dot{w} - \tilde{a4}uw - \tilde{a5}\sin\theta - \tilde{a6}\cos\theta) - k_\theta \operatorname{sgn}(s)) \tag{3.38}$$

Given bounds on the modeling errors $\tilde{a1}, \tilde{a2}, \tilde{a3}, \tilde{a4}, \tilde{a5}, \tilde{a6}$ we see that choosing k such that $k \geq |\tilde{a1}\ddot{\theta}_r + \tilde{a2}u\dot{\theta}_r + \tilde{a3}\dot{w} + \tilde{a4}uw + \tilde{a5}\sin\theta + \tilde{a6}\cos\theta| + \eta$ .With η positive, the sliding condition $\dot{V} \leq -\eta s$ is satisfied. This guarantees that the surface $s = 0$ is reached in a finite time and that once on the surface, the trajectories remain on the surface and therefore tend to θ_d exponentially.

3.2.2 Adaptive Controller Design for Pitch

Given a desired, bounded trajectory and with some parameter uncertainty, the adaptive controller design problem is to derive a control law for torque and estimation law for the parameter uncertainty such that $\theta_d \approx \theta$. Let us define $\tilde{a} = \hat{a} - a$ with a being a constant vector of uncertain parameters describing the vehicles mass, inertia and hydrodynamic properties and $\hat{a}$ is its estimate.

Consider a Lyapunov Function candidate $V = \frac{1}{2}a1s_\theta^2 + \frac{1}{2}\tilde{\mathbf{a}}^T\Gamma^{-1}\tilde{\mathbf{a}} > 0$ where Γ is a symmetric positive definite matrix. Differentiating and substituting from Eq(3.34) from the discussion above yields:

$$\dot{V} = s_\theta(\Delta e - a1\ddot{\theta}_r - a2u\dot{\theta}_r - a3\dot{w} - a4uw - a5\sin\theta - a6\cos\theta) + \dot{\tilde{\mathbf{a}}}^T\Gamma^{-1}\tilde{\mathbf{a}} \tag{3.39}$$

Defining a vector $\mathbf{Y}$ and $\mathbf{a}$ such that the terms are linearly dependent on $\mathbf{a}$.

Define the known matrix Y to be:

$$\mathbf{Y} = \begin{bmatrix} \ddot{\theta}_r & u\dot{\theta}_r & \dot{w} & uw & \sin\theta & \cos\theta \end{bmatrix} \tag{3.40}$$

so that,

$$\mathbf{a} = \left[a1\ a2\ a3\ a4\ a5\ a6\right]^T \tag{3.41}$$

resulting in

$$a1\ddot{\theta}_r + a2u\dot{\theta}_r + a3\dot{w} + a4uw + a5\sin\theta + a6\cos\theta = \mathbf{Ya} \tag{3.42}$$

Taking the control law to be

$$\Delta e = \mathbf{Y\hat{a}} - k_\theta s_\theta \tag{3.43}$$

which includes a "feedforward" term $\mathbf{Y\hat{a}}$ in addition to a simple PD term, $k_\theta s_\theta$, leads to

$$\dot{V} = s_\theta(\mathbf{Y\tilde{a}} - k_\theta s_\theta) + \dot{\tilde{\mathbf{a}}}^T \Gamma^{-1} \tilde{\mathbf{a}} \tag{3.44}$$

Updating the parameter estimates $\hat{\mathbf{a}}$ according to the correlation integrals

$$\dot{\hat{\mathbf{a}}} = -\Gamma \mathbf{Y}^T s_\theta \tag{3.45}$$

then yields

$$\dot{V} = -k_\theta s_\theta^2 \leq 0 \tag{3.46}$$

This implies that the output error converges to the surface $s = 0$ which, given the filter like definition of s shows that $\tilde{\theta}$ and $\dot{\tilde{\theta}}$ tend to 0 as t tends to infinity. Therefore the system global stability (boundedness of the vectors θ, $\dot{\theta}$ and $\hat{\mathbf{a}}$), and the tracking error convergence are guaranteed by the adaptive controller.

3.2.3 Limiting adaptation

Adding a dead-zone to shut adaptation off for small tracking errors to prevent noise and disturbances from causing parameter drift, the equations are modified as follows:

$$s_{\theta\Delta} = s_\theta - \Phi_\theta \, sat\left(\frac{s_\theta}{\Phi_\theta}\right) \tag{3.47}$$

$$V = \frac{1}{2} a1 s_{\theta\Delta}^2 + \frac{1}{2}\tilde{\mathbf{a}}^T \Gamma^{-1} \tilde{\mathbf{a}} > 0 \tag{3.48}$$

$$a1 > 0$$

$$\Gamma^{-1} = \Gamma^{-T} > 0$$

$$\tilde{\mathbf{a}} = \hat{\mathbf{a}} - \mathbf{a} \tag{3.49}$$

$$\dot{V} = s_{\theta\Delta} \, a1 \dot{s}_\theta + \dot{\tilde{\mathbf{a}}}^T \Gamma^{-1} \tilde{\mathbf{a}} \tag{3.50}$$

$$\dot{V} = s_{\theta\Delta} (\Delta e - \mathbf{Y}\mathbf{a}) + \dot{\tilde{\mathbf{a}}}^T \Gamma^{-1} \tilde{\mathbf{a}} \tag{3.51}$$

$$\Delta e = \mathbf{Y}\hat{\mathbf{a}} - k_\theta s_\theta \tag{3.52}$$

$$\dot{V} = s_{\theta\Delta} (\mathbf{Y}\hat{\mathbf{a}} - k_\theta s_\theta - \mathbf{Y}\mathbf{a}) + \dot{\tilde{\mathbf{a}}}^T \Gamma^{-1} \tilde{\mathbf{a}} = s_{\theta\Delta} (\mathbf{Y}\tilde{\mathbf{a}} - k_\theta s_\theta) + \dot{\tilde{\mathbf{a}}}^T \Gamma^{-1} \tilde{\mathbf{a}} \tag{3.53}$$

$$\dot{\hat{\mathbf{a}}} = -\Gamma \mathbf{Y}^T s_{\theta\Delta} = \dot{\tilde{\mathbf{a}}} \tag{3.54}$$

$$\dot{V} = (\dot{\tilde{\mathbf{a}}}^T \Gamma^{-1} + s_{\theta\Delta} \mathbf{Y})\tilde{\mathbf{a}} - s_{\theta\Delta} k_\theta s_\theta \tag{3.55}$$

$$\dot{V} = -s_{\theta\Delta} k_\theta s_\theta = -s_{\theta\Delta} k_\theta \left(s_{\theta\Delta} + \Phi_\theta \, sat\left(\frac{s_\theta}{\Phi_\theta}\right)\right) = -k_\theta (s_{\theta\Delta})^2 - k_\theta \Phi_\theta |s_{\theta\Delta}| \tag{3.56}$$

$$k_\theta \Phi_\theta \geq 0 \tag{3.57}$$

$$\dot{V} < -k_\theta (s_{\theta\Delta})^2 < 0 \tag{3.58}$$

3.3 Depth

Looking first at the equations for rigid body dynamics we know that

$$\dot{z} = w\cos\theta - u\sin\theta \tag{3.59}$$

$$\ddot{z} = \dot{w}\cos\theta - w\sin\theta\dot{\theta} - \dot{u}\sin\theta - u\cos\theta\dot{\theta} \tag{3.60}$$

Grouping terms, we can write

$$\ddot{z} = [\dot{w}\cos\theta - \dot{u}\sin\theta] + [-w\sin\theta - u\cos\theta]\dot{\theta} \tag{3.61}$$

$$\ddot{z} = f(\theta) + b(\theta)u \tag{3.62}$$

From these equations, it is seen that depth control and pitch control are inter-related. Altering pitch is the only method of changing or maintaining depth with a single set of elevators. Using the same methodology the following terms are defined.

$$\tilde{z} = z - z_d \tag{3.63}$$

$$\dot{\tilde{z}} = \dot{z} - \dot{z}_d \tag{3.64}$$

$$\ddot{\tilde{z}} = \ddot{z} - \ddot{z}_d \tag{3.65}$$

Here a reference velocity, $\dot{z}_r$, is formed by shifting the desired velocity, $\dot{z}_d$, according to the position error, $\tilde{z}$.

$$\dot{z}_r = \dot{z}_d - \lambda_z \tilde{z} \tag{3.66}$$

$$s_z = \dot{\tilde{z}} + \lambda_z \tilde{z} = \dot{z} - \dot{z}_r \tag{3.67}$$

$$\ddot{z}_r = \ddot{z}_d - \lambda_z \dot{\tilde{z}} \tag{3.68}$$

$$\dot{s}_z = \ddot{z} - \ddot{z}_r \tag{3.69}$$

Choosing a control law

$$\dot{s} = \ddot{z} - \ddot{z}_d + \lambda_z \dot{\tilde{z}} = f + bu - \ddot{z}_d + \lambda_z \dot{\tilde{z}} \tag{3.70}$$

$$\hat{u} = -\hat{f} + \ddot{z}_d - \lambda_z \dot{\tilde{z}} \tag{3.71}$$

$$u = b^{-1}[\hat{u} - k_z s_z] \tag{3.72}$$

The depth input, $\dot{\theta}_d$, is a quantity used by the pitch controller. It is a relatively straight forward matter to back difference and integrate $\dot{\theta}_d$ to get the remaining two quantities, θ_d and $\ddot{\theta}_d$, used by the pitch controller. What remains is to select an appropriate value of λ_z and λ_θ. With the end result of the pitch sliding controller being that pitch behaves as a linear system, it is straight forward to make the restriction that $\lambda_z \leq \lambda_\theta / 4$. Because the outer depth loop generates the θ_d, $\dot{\theta}_d$, and $\ddot{\theta}_d$, maintaining a large difference between the two bandwidths, allows the pitch and depth loop dynamics to be evaluated independently and without adverse excitation between the two. The controller equations

are presented in the Appendix.

3.4 Yaw

Taking the moment equation for yaw the robust sliding controller is derived as follows:

$$\begin{bmatrix} mx_G + x_{ma}Mav & Jzz + Jamz & 0 \end{bmatrix} \begin{bmatrix} \dot{v} \\ \dot{r} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} NuvU - C_{fin}Ux_{fin} & NurU - mUx_G - C_{fin}Ux_{fin}^2 & 0 \end{bmatrix} \begin{bmatrix} v \\ r \\ \varphi \end{bmatrix} \quad (3.73)$$

$$+ \begin{bmatrix} x_{fin}C_{fin}U^2 \end{bmatrix} \begin{bmatrix} \delta r \end{bmatrix}$$

$$(mx_G + x_{ma}Mav)\dot{v} + (Jzz + Jamz)\dot{r} = (NuvU - C_{fin}Ux_{fin})v + (NurU - mUx_G - C_{fin}Ux_{fin}^2)r \quad (3.74)$$

$$+ (x_{fin}C_{fin}U^2)\delta r$$

Arranging the state variables in a linear combination

$$u = U \quad (3.75)$$

$$a1 = Jzz + Jamz \quad (3.76)$$

$$a2 = mx_G + x_{ma}Mav \quad (3.77)$$

$$a3 = Nur - mx_G - C_{fin}x_{fin}^2 \quad (3.78)$$

$$a4 = Nuv - C_{fin}x_{fin} \quad (3.79)$$

$$a1\ddot{\psi} + a2\dot{v} + a3u\dot{\psi} + a4uv = -x_{fin}C_{finu}{}^2\delta r = \Delta r \tag{3.80}$$

Defining the following terms:

$$\tilde{\psi} = \psi - \psi_d \tag{3.81}$$

Here a reference velocity, $\dot{\psi}_r$, is formed by shifting the desired velocity, $\dot{\psi}_d$, according to the position error, $\tilde{\psi}$.

$$\dot{\psi}_r = \dot{\psi}_d - \lambda_\psi \tilde{\psi} \tag{3.82}$$

$$s_\psi = \dot{\tilde{\psi}} + \lambda_\psi \tilde{\psi} = \dot{\psi} - \dot{\psi}_r \tag{3.83}$$

$$\dot{s_\psi} = \ddot{\psi} - \ddot{\psi}_r \tag{3.84}$$

Adding a dead-zone to shut adaptation off for small tracking errors to prevent noise and disturbances from causing parameter drift, the equations are modified as follows:

$$s_{\psi\Delta} = s_\psi - \Phi_\psi sat\left(\frac{s_\psi}{\Phi_\psi}\right) \tag{3.85}$$

Define a Lyapunov function candidate, V(t), a positive definite function:

$$V = \frac{1}{2}a1s_{\psi\Delta}^2 + \frac{1}{2}\tilde{\mathbf{a}}^T\Gamma^{-1}\tilde{\mathbf{a}} > 0 \tag{3.86}$$

$$a1 > 0 \quad \Gamma^{-1} = \Gamma^{-T} > 0 \qquad \tilde{\mathbf{a}} = \hat{\mathbf{a}} - \mathbf{a}$$

Differentiating,

$$\dot{V} = s_{\psi\Delta}\, a1\dot{s_\psi} + \dot{\tilde{\mathbf{a}}}^T\Gamma^{-1}\tilde{\mathbf{a}} \tag{3.87}$$

Defining a vector $\mathbf{Y}$ and $\mathbf{a}$ such that the terms are linearly dependent on $\mathbf{a}$. Define the known matrix Y to be:

$$\mathbf{Y} = \left[\ddot{\psi}_r \; \dot{v} \; u\dot{\psi}_r \; uv\right] \tag{3.88}$$

so that,

$$\mathbf{a} = \left[a1 \; a2 \; a3 \; a4\right]^T \tag{3.89}$$

resulting in

$$a1\dot{s_\psi} = a1(\ddot{\psi} - \ddot{\psi}_r) = \Delta e - \mathbf{Ya} \tag{3.90}$$

$$\dot{V} = s_{\psi\Delta}(\Delta e - \mathbf{Ya}) + \dot{\tilde{\mathbf{a}}}^T \Gamma^{-1} \tilde{\mathbf{a}} \tag{3.91}$$

Taking the control law to be

$$\Delta e = \mathbf{Y}\hat{\mathbf{a}} - k_\psi s_\psi \tag{3.92}$$

$$k_\psi > 0$$

which includes a "feedforward" term $\mathbf{Y}\hat{\mathbf{a}}$ in addition to a simple PD term, $k_\psi s_\psi$, leads to

$$\dot{V} = s_{\psi\Delta}(\mathbf{Y}\hat{\mathbf{a}} - k_\psi s_\psi - \mathbf{Ya}) + \dot{\tilde{\mathbf{a}}}^T \Gamma^{-1} \tilde{\mathbf{a}} = s_{\psi\Delta}(\mathbf{Y}\tilde{\mathbf{a}} - k_\psi s_\psi) + \dot{\tilde{\mathbf{a}}}^T \Gamma^{-1} \tilde{\mathbf{a}} \tag{3.93}$$

Updating the parameter estimates $\hat{\mathbf{a}}$ according to the correlation integrals

$$\dot{\hat{\mathbf{a}}} = -\Gamma \mathbf{Y}^T s_{\psi\Delta} = \dot{\tilde{\mathbf{a}}} \tag{3.94}$$

$$\dot{V} = (\dot{\tilde{\mathbf{a}}}^T \Gamma^{-1} + s_{\psi\Delta}\mathbf{Y})\tilde{\mathbf{a}} - s_{\psi\Delta} k_\psi s_\psi \tag{3.95}$$

$$\dot{V} = -s_{\psi\Delta} k_\psi s_\psi = -s_{\psi\Delta} k_\psi \left(s_{\psi\Delta} + \Phi_\psi \, sat\left(\frac{s_\psi}{\Phi_\psi}\right)\right) = -k_\psi (s_{\psi\Delta})^2 + k_\psi \Phi_\psi |s_{\psi\Delta}| \tag{3.96}$$

$$k_\psi \Phi_\psi \geq 0 \tag{3.97}$$

$$\dot{V} < -k_\psi (s_{\psi\Delta})^2 < 0 \tag{3.98}$$

If u and w are bounded, then $V \geq 0$ and $\dot{V} < 0$ which means V is bounded. Thus $s_\psi, \tilde{\mathbf{a}}$ are bounded$\tilde{\mathbf{a}}$. being bounded implies $\hat{\mathbf{a}}$ is boundeds_ψ. being bounded implies $\tilde{\psi}, \dot{\tilde{\psi}}$ are bounded. Assuming that $\psi_d, \dot{\psi}_d$ are bounded then $\psi, \dot{\psi}$ are bounded. For positive values of u, with Δr bounded then $\ddot{\psi}$ is bounded. Assuming that $\ddot{\psi}_d$ is bounded, then $\dot{s}$ is bounded and $\ddot{V}$ is bounded so $\dot{V} \to 0$ by Barbalat's Lemma. So $s_{\psi\Delta} \to 0$.

Equations used in the controller are in the Appendix.

3.5 Summary

Model based adaptive sliding controllers for depth/pitch and yaw are designed using the uncoupled models developed in Chapter 2. Model parameter adaptation is used to accommodate slowly changing parameters and to ensure tracking and stability in the presence of modeling errors and disturbances.

Chapter 4

Control Implementation and Simulation

4.1 Implementation

Chapter 4 discusses several controller implementation details. After state estimation and the trajectory generator, the other implementation details can be considered fine tuning promoting robust variable speed performance while minimizing actuator motion and power consumption. A model and trajectory generator that compensate for slowing and the resulting reduced fin and body lift forestall the onset of stall conditions. Compensating for this reduced lift, allows the vehicle to achieve a less aggressive trajectory without becoming uncontrollable in depth or losing steerageway. The actuation filter, based on the physical gear backlash, dramatically reduces fin motion. Model parameter adaptation improves tracking performance by continually refining the model estimates.

4.2 State Estimation

A number of states utilized by this controller are not measurable with the current vehicle instrumentation. The states selected are the minimum necessary to capture the relevant vehicle dynamic instabilities. It is necessary to either improve the instrumentation suite to provide the desired parameters or estimate them with some sort of observer. The measured vehicle state is $\hat{X}_{meas} = \left[p\ q\ r\ z\ \phi\ \theta\ \psi\right]^T$. The missing parameters, $\hat{X}_{\text{missing}} = \left[u\ v\ w\right]^T$, are necessary for controller implementation. Laboratory simulations use a full state nonlinear model for vehicle dynamics. A vehicle observer using an adaptation of that simulation calculates missing vehicle parameters. Measured states "reset" the

corresponding observer parameters every control cycle so that calculated parameters remain reasonable. A cost of using this technique is the added computation load on the vehicle computer for control purposes. The number of iterations in the observer is reduced from the laboratory simulator to prevent impacting the vehicle 200 msec control cycle time. This does not adversely affect the calculated parameters.

A preferred approach to state estimation is an instrumentation package providing the desired parameters, reducing the computation load. Installation of a Sontek acoustic doppler velocimeter (ADV) is planned.

4.3 Speed Dependent Pitch Limit

When trying to determine the significant operating parameters during slow speed operations, one approach is to examine speed's role in the dynamic model. The most direct effects are reduced body and fin lift which behave according to:

$$Lf = 0.5\rho A(Cl)U^2\delta \tag{4.1}$$

Slower operating speeds drastically affect depth/pitch. Longer duration transients result from the reduced lift and torque generated by the fins and body. In the pitch plane, there are two added effects. With the elevator torque having to counter the CG-CB moment arm, the elevators are unable to sustain as large a pitch angle at slower speeds. Positive buoyancy limits depth/pitch control the most. When speed drops to the point where hull lift is insufficient to counter the buoyancy, depth becomes uncontrollable.

As speed changes, pitch angle limits change as listed below with K=26 and k=9.

$$\theta_{limit} = KU - k \tag{4.2}$$

While varying speed, this limit keeps the desired trajectory physically realizable and attempts to avoid stall conditions at slower speeds.

4.4 Speed Compensated Trajectory Generator

A trapezoidal trajectory generator illustrated in Figure 4.1, using physically realizable accelerations and velocities, generates a desired trajectory for heading and depth changes. These trajectory states, with the vehicle dynamics model, generate "feedforward" terms for the control input to attain low tracking error. The feedback terms, while working to keep the tracking error zero, are only compensating for model errors.

With slower operating speeds, the acceleration and velocities obtainable are smaller. Trajectory accelerations and velocities are reduced accordingly and the correction magnitudes for position and velocity errors are greatly reduced. The following relations limit trajectory accelerations and velocities as a speed function:

$$\dot{\psi}_{max} = U_{desired} \cdot \dot{\psi}_{max} \tag{4.3}$$

$$\dot{Z}_{max} = U_{desired} \cdot \sin(\theta_{limit}) \tag{4.4}$$

$$\ddot{\psi}_{max} = 2 \cdot U_{desired} \tag{4.5}$$

$$\ddot{Z}_{max} = 0.01 \cdot U_{desired} \tag{4.6}$$

Vehicle open loop performance at varying speeds[11] determined these values. The limits are within the vehicle physical limitations so during a transient, full fin deflection is not required to achieve an impossible desired yaw rate or pitch angle and some actuator range is available for corrections.

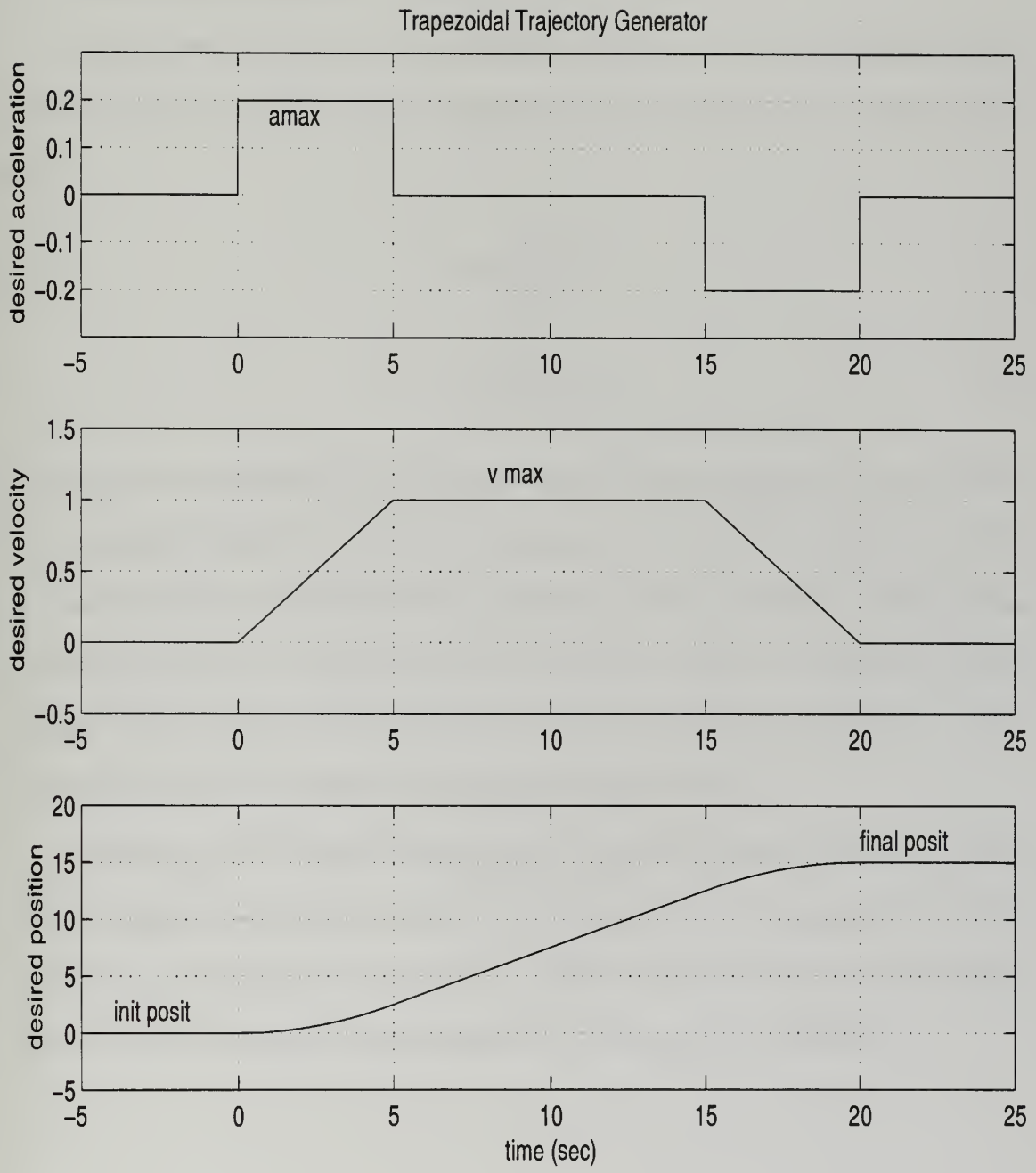

Figure 4.1 Trapezoidal Trajectory Generator. A physically realizable trajectory is generated using realistic values taken from vehicle open loop performance. The maximum acceleration and velocity decreases with speed to compensate for reduced lift.

4.5 Minimizing Actuator Motions

With control surface torque inputs directly dependent on speed, the larger of estimated vehicle speed or the desired speed determines fin position for a specific control input.

$$Lf = 0.5\rho A(Cl)U^2\delta \tag{4.7}$$

$$\uparrow \delta = \frac{Lf}{0.5\rho A(Cl)U^2} \downarrow \tag{4.8}$$

This approach eliminates excessive actuator motion when the vehicle is slowing, preventing unmodeled dynamics excitation and unnecessary power drain. Additionally, during start-up transients or speed increases, the initial fin motions are smaller, reducing fin drag, allowing the vehicle to reach operating speed quicker. Figure 4.2 compares two depth changes while slowing. In the transient plotted with broken lines, excessive actuator motion drops speed until stall occurs at approximately t=213,G50

degrading pitch and depth keeping. Using a speed trajectory generator to create a smooth transition between states would be an alternate but more computationally costly approach to minimize actuator motion. This example demonstrates how excessive actuator motion driven by any source increases drag and reduces propulsion efficiency.

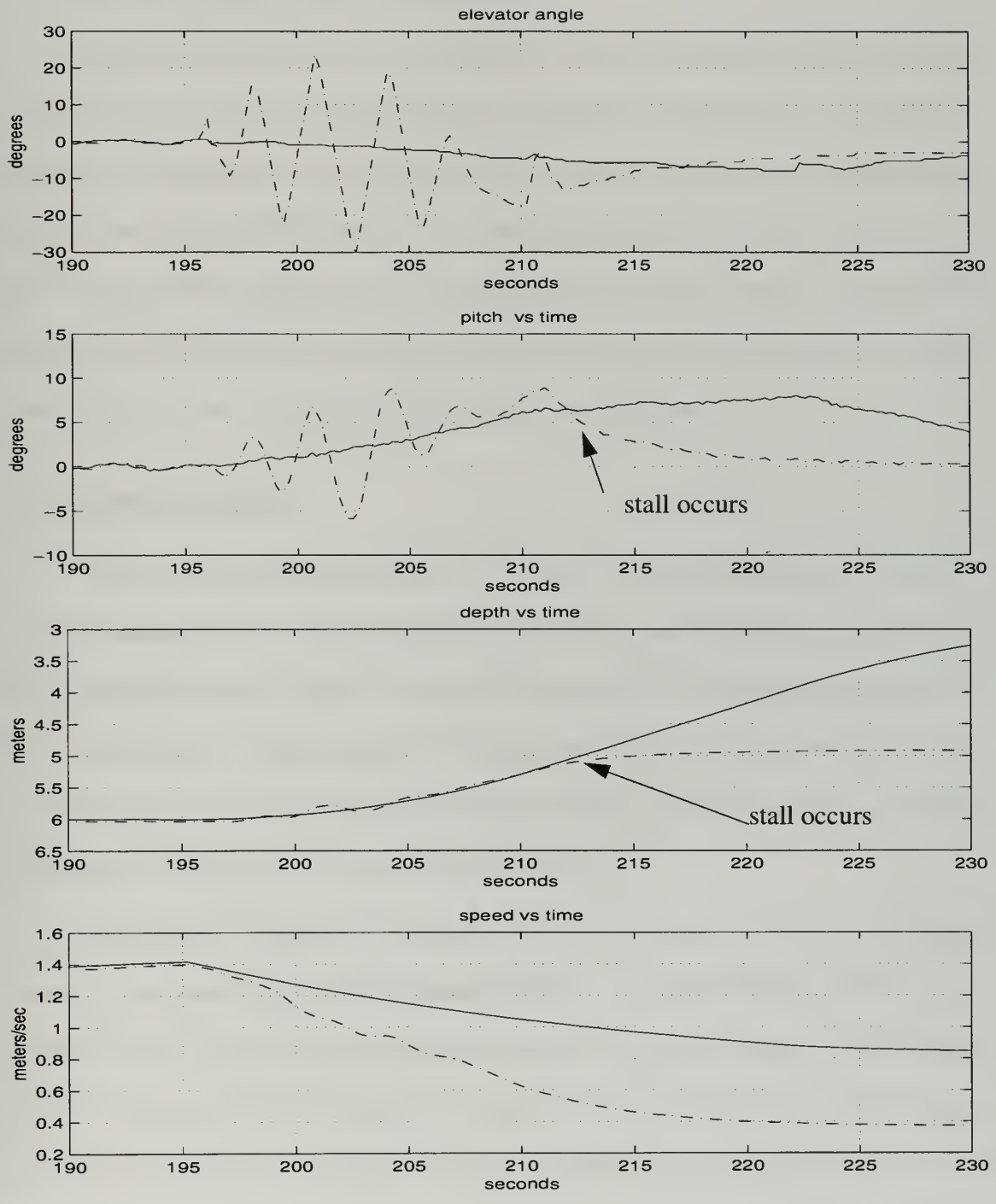

Figure 4.2 Comparing Performance With and Without Limited Elevator Motion. With elevator position based on a lower commanded speed, the controller over corrects for pitch errors. Large elevator oscillations increase fin drag and slow the vehicle. As the vehicle slows, hull lift is lost, stall occurs and poor depth keeping results.

4.6 Robust Sliding Control Simulation

After implementing all of the controller features discussed, vehicle performance simulations aid overall and individual feature evaluation. The first task is to evaluate performance with multiple speeds during the same mission. The mission profile used is a launch from zero speed to a maximum 1.5 meters/sec and then two step decreases to 0.7 meters/sec. Heading changes occur simultaneously with speed changes. Next, actuation filtering benefits are shown with a mission profile comparison. The vehicle model used for simulations has fixed vertical fins added as discussed in Chapter 5.

4.6.1 Speed Changes

Figure 4.3 shows heading and depth performance with speed changes at 180 seconds and 360 seconds. The corresponding change in heading and depth rates are most easily seen as longer transients. Figure 4.4 expands the original time scale where actual and trajectory heading are indistinguishable. The pitch and depth control performance is not as crisp and precise as heading because of pitch and depth inter-dependency. A pitch and depth control expanded view is also shown in Figure 4.4.

Because the "feed forward" control input terms are generated based on the desired trajectory dynamics instead of accumulated errors, as in a typical PID controller, the corrections that are need to keep tracking errors small are greatly reduced. Acknowledging that the sliding controller feedback is not multi-variable, the feed forward terms do provide a multi-variable control input. This multi-variable control input is not in the form generally used in multi-variable controllers but the basic tenets are there. Individual feed forward input magnitudes as well as the role feedback plays in correcting for modeling errors is plotted in an overall and expanded mission view in Figure 4.5 and Figure 4.6.

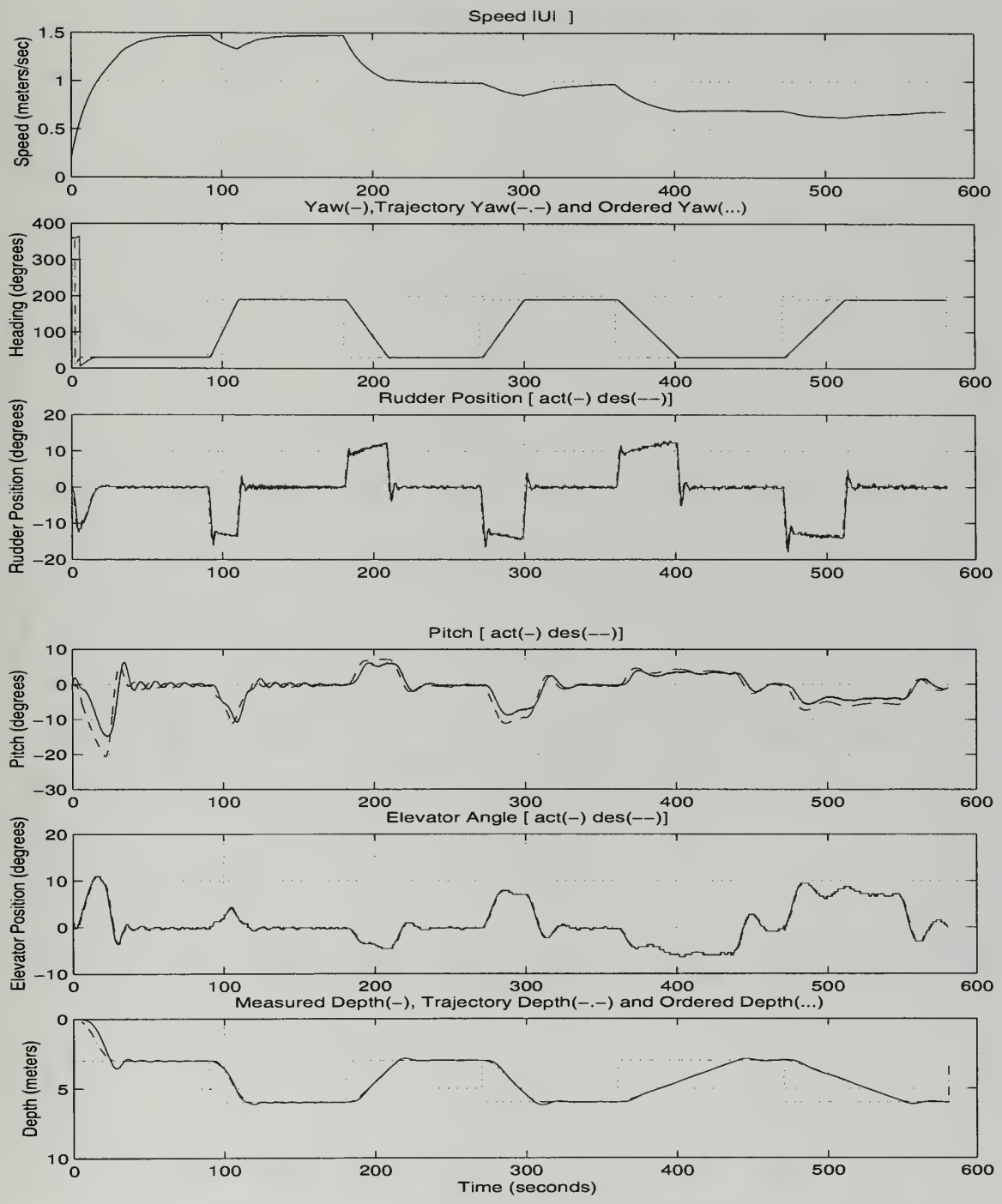

Figure 4.3 Performance With Varying Speed. Transients at lower speeds require longer times to complete as control surface lift is less. Despite large changes in forward speed vehicle trajectory tracking is excellent.

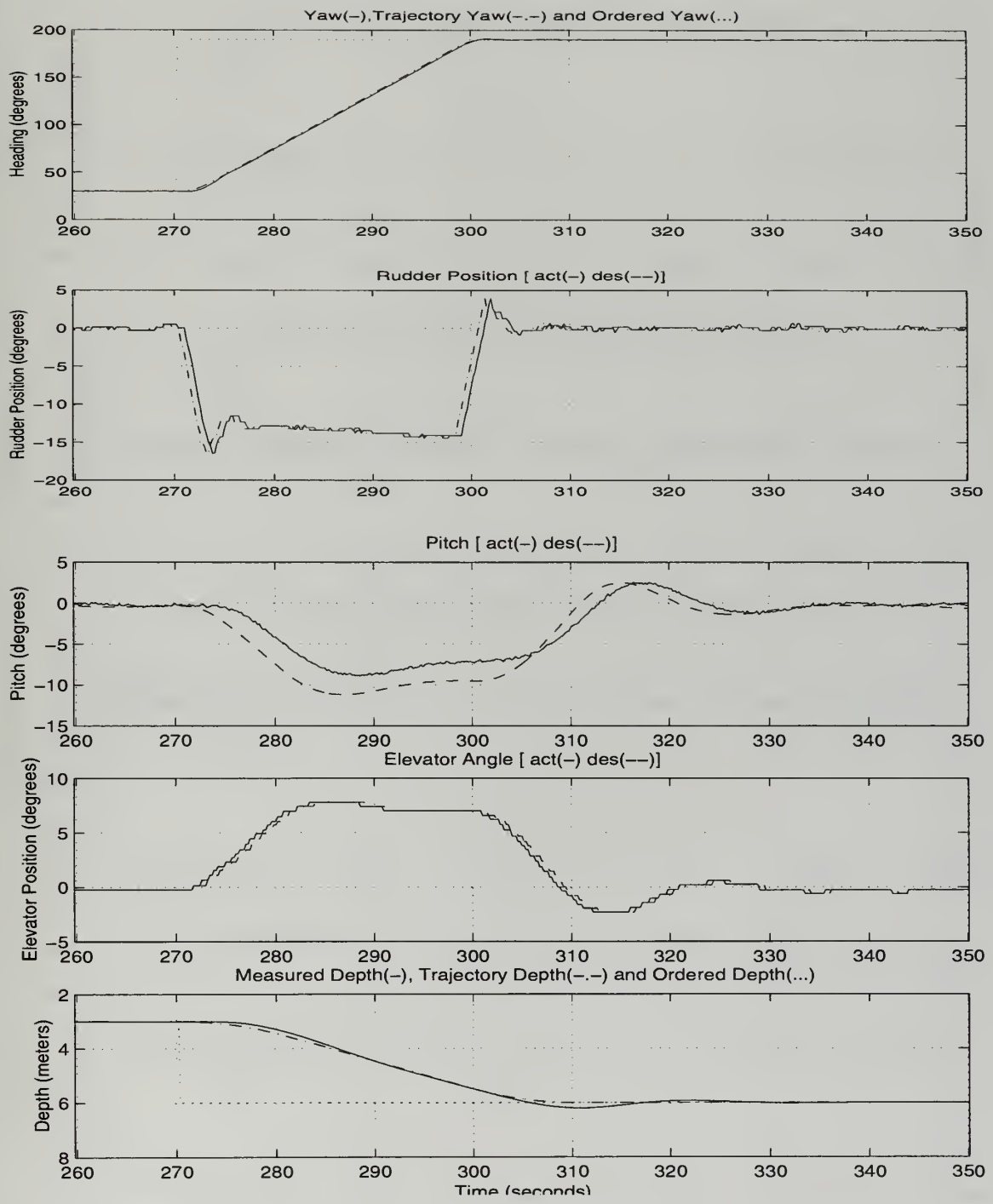

Figure 4.4 Expanded View of Figure 4.3- transient at 1 m/sec. One transient is examined closely for a better look at trajectory tracking. Heading and depth are difficult to separate from the trajectory in the plots. Pitch control is not as crisp and precise as heading because of the inter-dependency of pitch and depth.

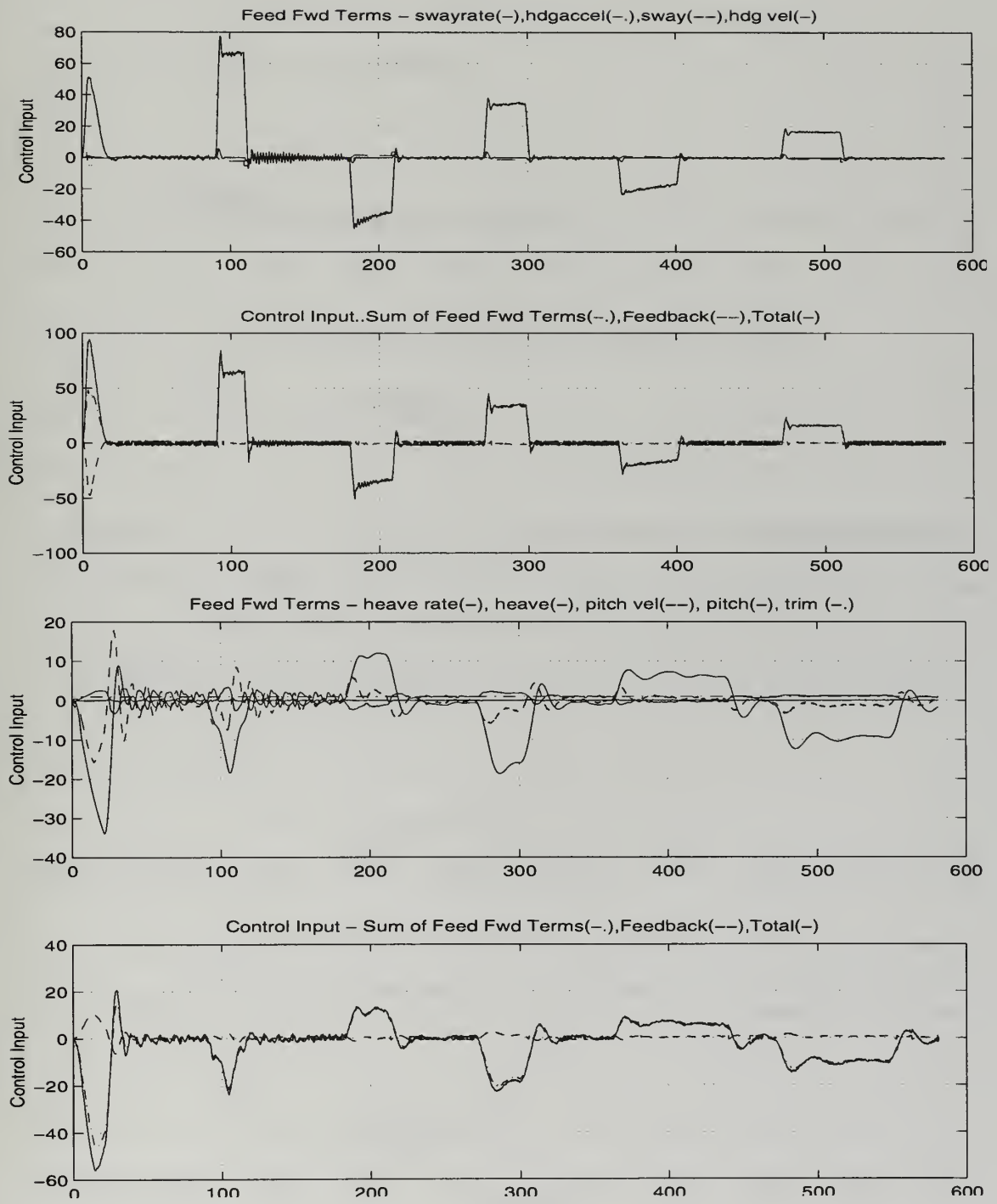
Feed Fwd Terms – swayrate(–),hdgaccel(–.),sway(––),hdg vel(–)
Control Input
Control Input..Sum of Feed Fwd Terms(–.),Feedback(––),Total(–)
Control Input
Feed Fwd Terms – heave rate(–), heave(–), pitch vel(––), pitch(–), trim (–.)
Control Input
Control Input – Sum of Feed Fwd Terms(–.),Feedback(––),Total(–)
Control Input

Figure 4.5 Control Inputs for Mission in Figure 4.3. Individual feed forward terms and their sum are plotted with feedback for comparison. Individual feed forward terms are difficult to see in this plot because of the time scale. During the maneuvers, feed forward terms are primarily responsible for tracking performance as feedback is small.

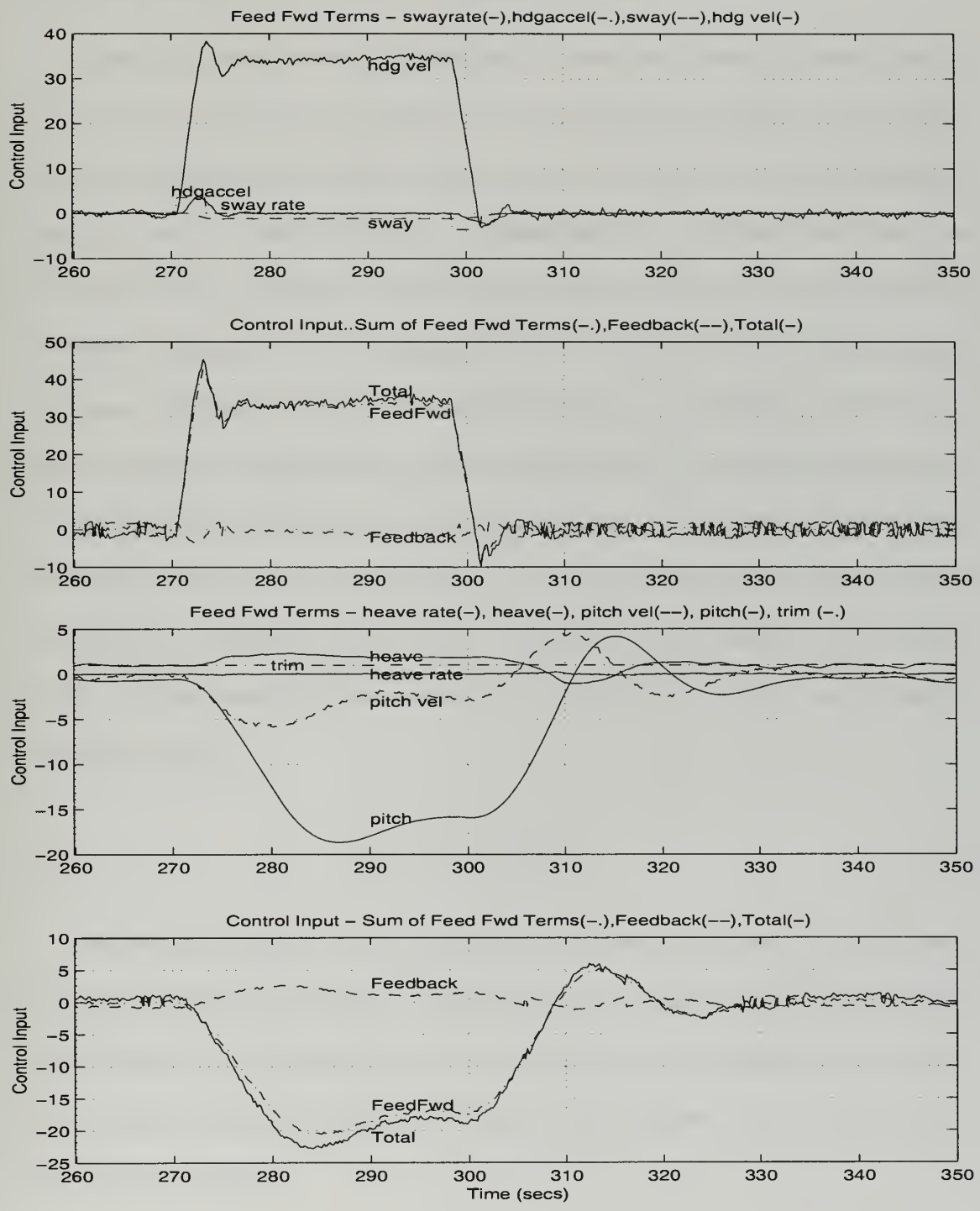

Figure 4.6 Control Inputs for Transient in Figure 4.4- transient at 1 m/sec. A closer look shows the relative importance of each feed forward term and the relatively small role feedback is playing in the tracking performance.

4.6.2 Eliminating Noise Driven Actuator Motion

Sensor noise driving excessive fin motion is a significant concern. Control activity at 5 Hz, the basic control cycle frequency, is an unnecessary power drain on the vehicle. Using alternate output modes built in to the sensor to reduce noise were unsuccessful and changing sensors is fiscally prohibitive. Considering the already heavy state estimator computational load, a rudimentary filter utilizes physical limitations. Gear backlash in the fin actuators is estimated to be 0.5 degrees and total actuator drivetrain backlash measures a minimum 5 degrees. Limiting actuator repositioning to increments greater than 0.5 degrees, effectively eliminates sensor noise during simulation with minimal performance cost. The only real loss in performance is during actuator positioning that further advances the fin and does not involve "gear slop". Fin activity decreases from 5 Hz to approximately 1 Hz with very satisfactory performance. Increased lift forces make limiting actuator motion more detrimental at higher speeds. The threshold for fin motion varies inversely with speed as follows:

$$\delta_{allowed} = 0.5 \cdot \frac{\pi}{180} \cdot \frac{1}{U} \tag{4.9}$$

The next figures show how the actuator filter reduces motion without adversely impacting performance. In Figure 4.7, with both missions plotted on the same axes, there is little difference on the large scale as well as the Figure 4.8 expanded scale. It is necessary to examine expanded transients in Figure 4.9 and Figure 4.10 to see the benefits gained in actuator motion.

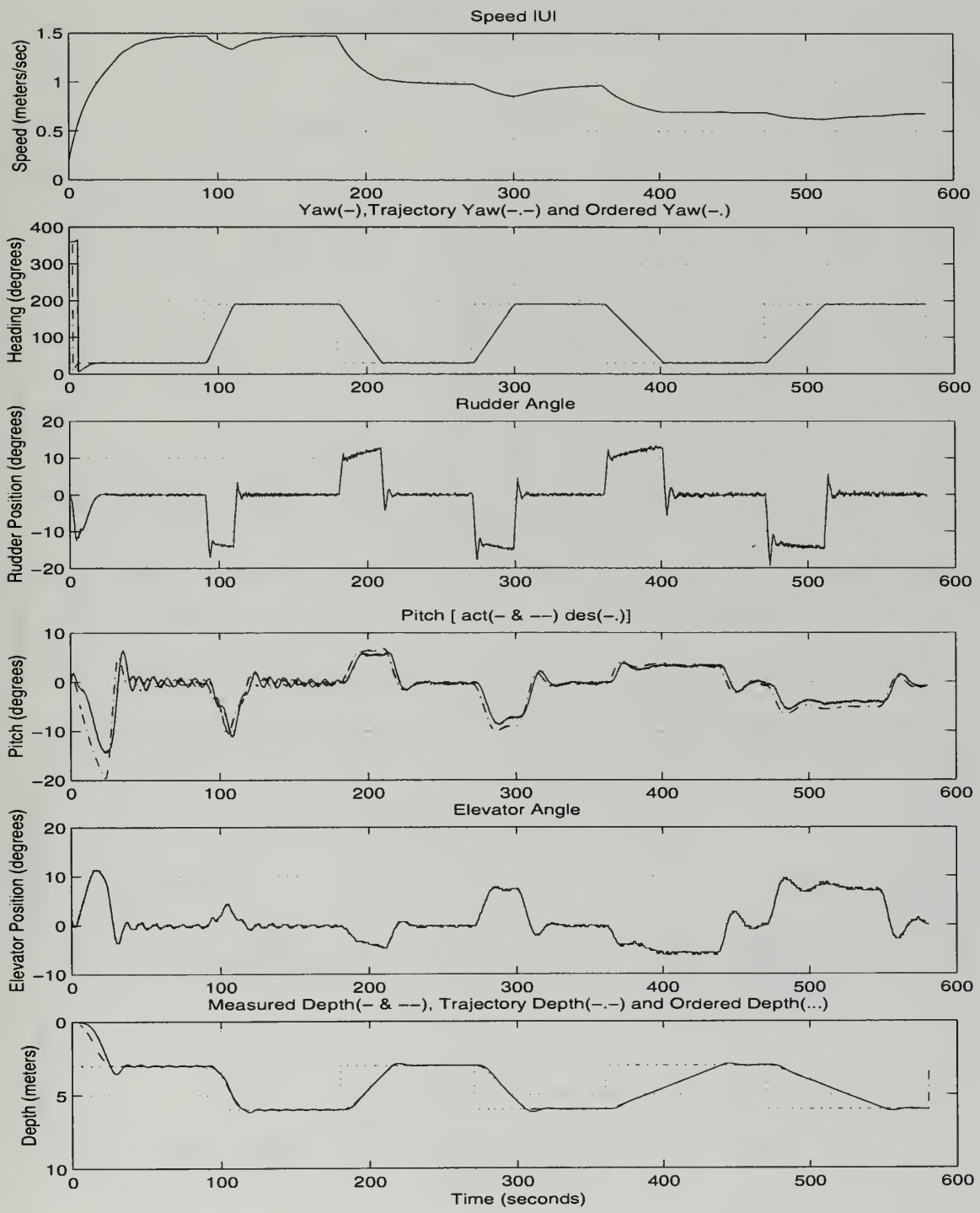

Figure 4.7 Comparing Performance With and Without Actuator Filter. Separating the plots of two missions on this time scale is difficult because the performance is essentially identical. A closer look is taken in the following figures.

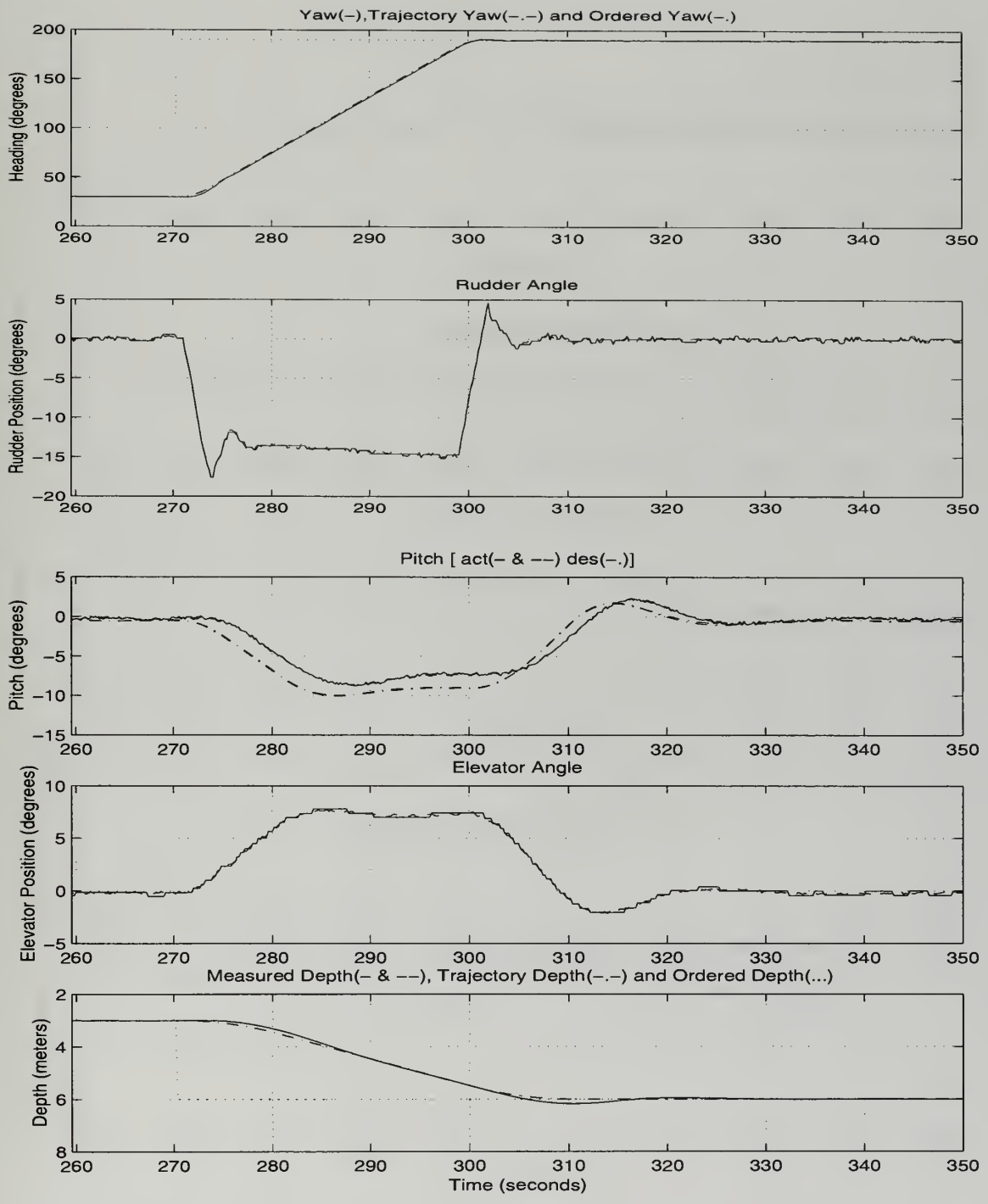

Figure 4.8 Expanded Portion of Actuator Filter Comparison - one transient at 1 m/ sec. Again, two plots are difficult to see when looking at only one transient, because of the virtually identical performance. This transient is expanded again in the next two plots.

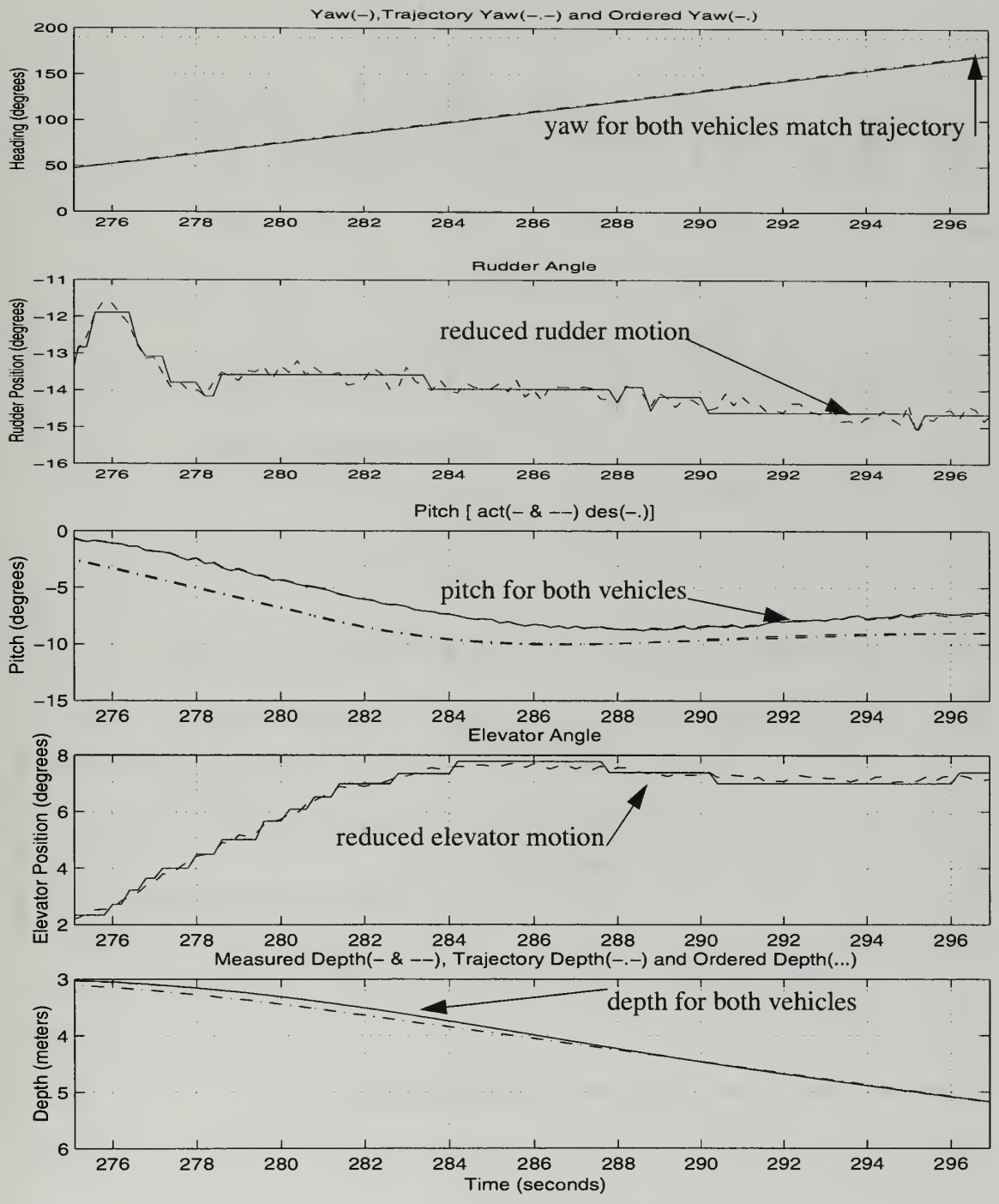

Figure 4.9 Expanded Portion of Actuator Filter Comparison - beginning of transient. Here the difference between trajectory and actual depth and heading can be seen but the two vehicle plots are essentially identical. What can be seen is just how much less control surface activity is needed for the same tracking performance.

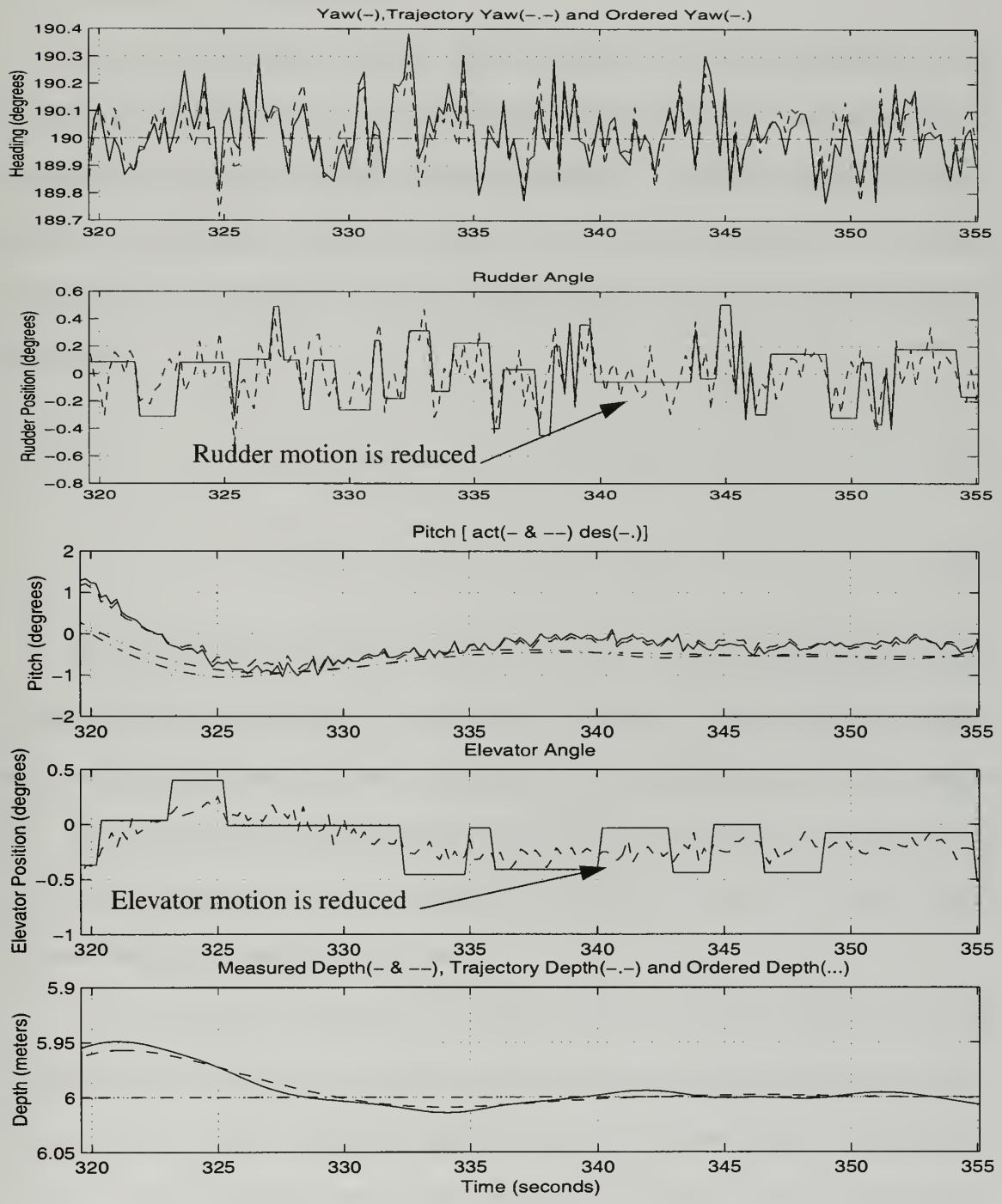

Figure 4.10 Expanded Portion of Actuator Filter Comparison - end of transient. As the new commanded heading and depth are reached, less control surface activity still gives good tracking performance. Performance initially may seem poor because of the vertical axis scaling.

4.6.3 Adaptive Sliding Control

Improving performance over time is the benchmark for evaluating adaptation. If the feedback magnitude decreases over time then the adaptation scheme works. Empirically selected gains allow slow adaptation to avoid oscillations. The gains are kept high to allow parameter estimations to slowly converge to the model parameters for a continual improvement in performance. The gain matrixes used for adaptation are given in (4.10) and (4.11).

$$\Gamma_{\psi} = \begin{bmatrix} 0.05 & 0.3 & 0.05 & 0.05 & 0.05 \end{bmatrix} \tag{4.10}$$

$$\Gamma_{\theta} = \begin{bmatrix} 0.4 & 0.4 & 0.4 & 0.04 & 0.1 & 0.04 \end{bmatrix} \tag{4.11}$$

Adapted parameter bounds of +/-30 percent from initial estimated values prevent instability due to parameter drift or unmodeled disturbances. This attempts to guarantee some minimum performance level. Adaptation critics complain about run away adaptation. With an accurate initial hydrodynamic parameter determination or estimation, and then limiting subsequent physical alterations, excellent performance is possible without excessive adaptation causing instabilities.

For the adaptation comparisons, heading and depth changes are made at a constant speed. The first simulations use updated model parameters with fixed vertical fins present to increase the vehicle's directional stability. The basis for these fins is the subject of Chapter 5. Initial parameters used by the controller differ from those used in the full state simulation by 30%, forcing some modeling errors. Even with these errors, feedback magnitude is insufficient for evaluation. To introduce additional errors, the model parameters

used by the controller are those for the vehicle without the vertical fixed fins while the simulation model has the fins in place. The model errors are then large enough for convenient comparisons and are shown in Figure 4.11 through Figure 4.15. Here again, the difference in performance is difficult to discern on a large scale. Figure 4.14 and Figure 4.15 expands the heading change at t = 90 seconds for a closer look.

Figure 4.15 isolates the feedback during three turns in the same direction, at the same speed for examination. The dashed plot shows the feedback without adaptation taking place. The solid plot presents the decrease in feedback with adaptation present and indicates adaptation is improving performance. The lower plot in Figure 4.15 shows a feedback close-up during the three turns.

4.7 Summary

Model based adaptive sliding controller implementation with a full state nonlinear estimator provides feed forward control inputs that allow accurate desired trajectory tracking. Using the detailed controller model and a trajectory generator that compensates for speed, excellent performance throughout the vehicle speed range is possible without controller retuning. Feed forward terms generated by the trajectory generator and vehicle state estimator achieve high tracking accuracy. These feed forward terms provide multi-variable control inputs based on the desired flight path rather than error accumulation. Additionally, an actuation filter prevents actuator motion of a smaller magnitude than the observed actuator drivetrain gear backlash. Any impact on performance using this filtering approach is minimal and actuator motion is reduced from 5 Hz to at least 1 Hz. Model parameter adaptation allows the continuing performance improvement as time progresses by refining the parameter estimates. These revised parameter estimates and the resulting performance

improvement manifest themselves as a reduction in the overall feedback required to compensate for modeling errors. The net gain with this controller is improved performance over a range of operating speeds with less actuator motion required.

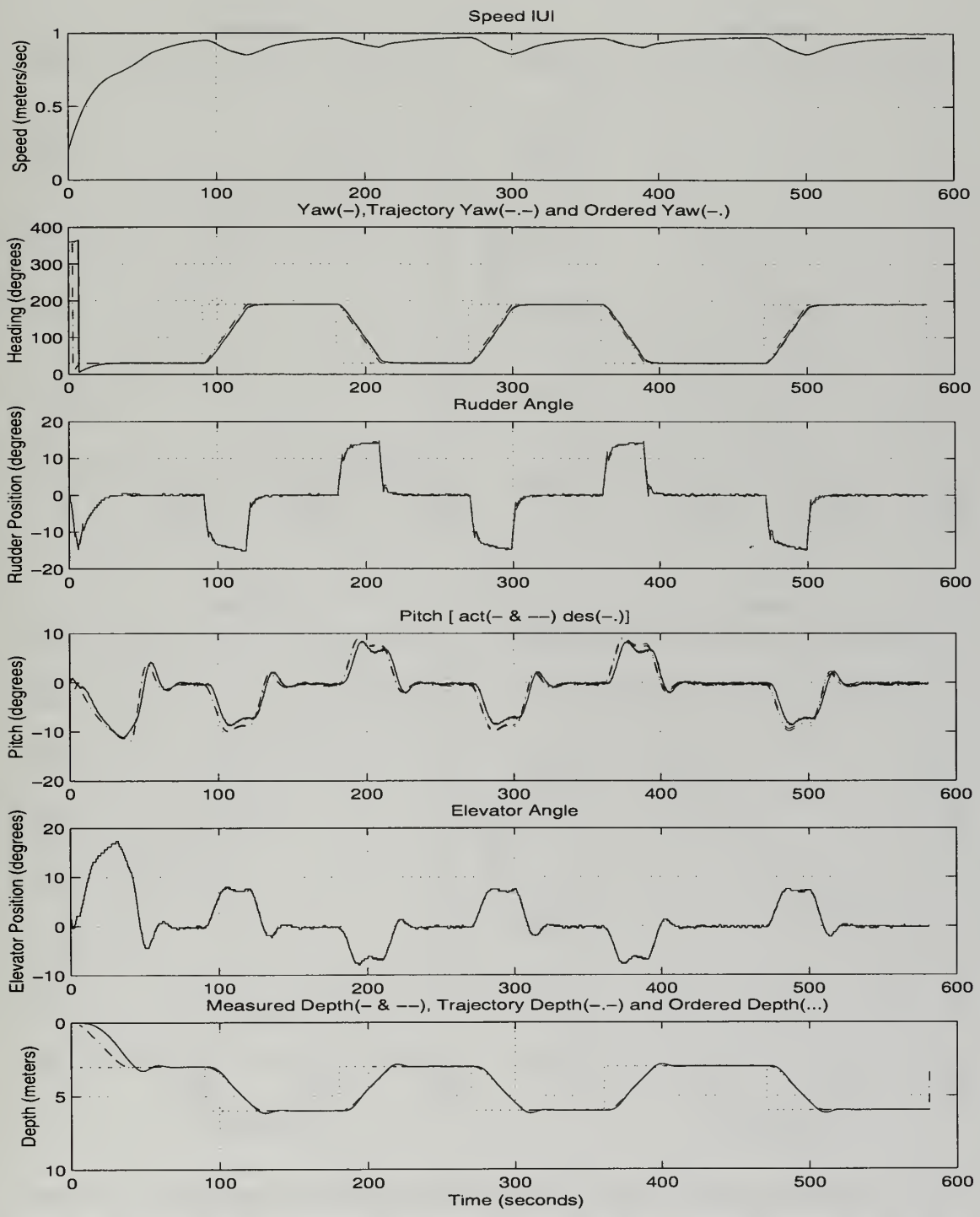

Figure 4.11 Comparison of Performance With and Without Adaptation. On this time scale the two vehicle plots can not be seen as the trajectories are essentially identical. Figure 4.14 shows a closer look.

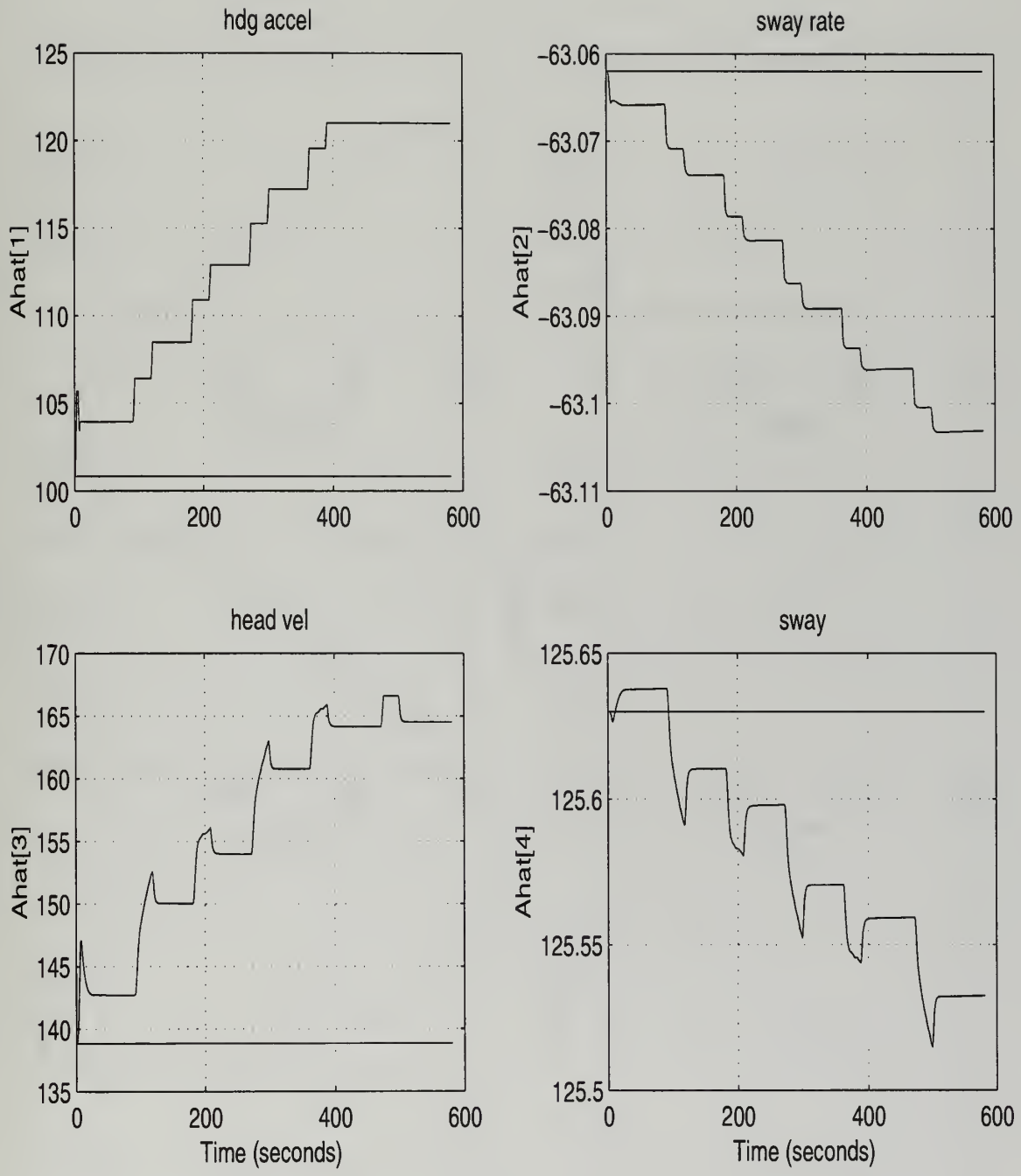

Figure 4.12 Adaptation of Heading Model Terms. Original parameters used by the controller are plotted with the updated estimates to show how adaptation changes the parameter values. These plots show a gradual convergence of the estimates. Note that excellent tracking performance is achieved with modeling errors present. The large feedback input in Figure 4.14 is an indication of the large modeling errors present.

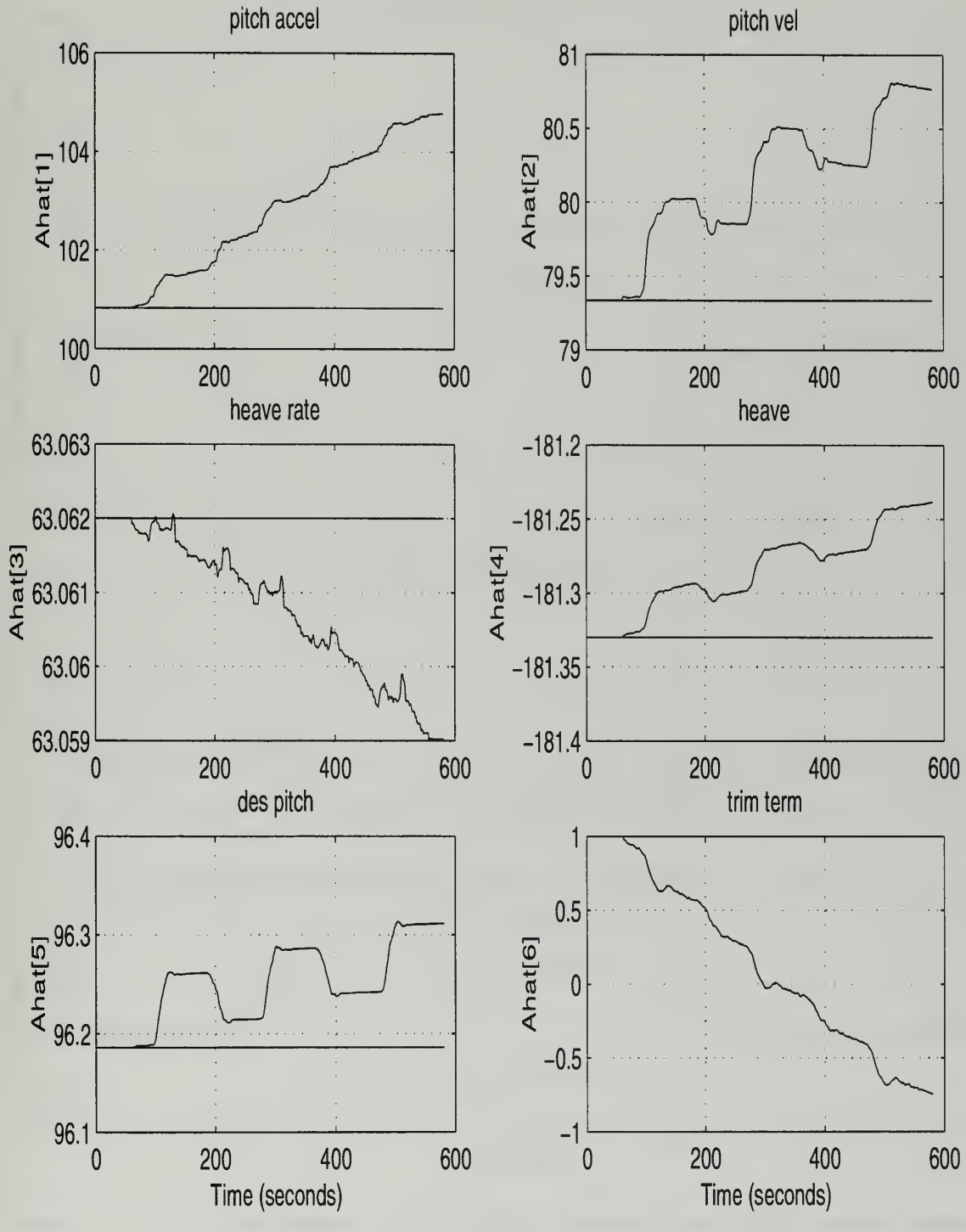

Figure 4.13 Adaptation of Pitch Model Terms.

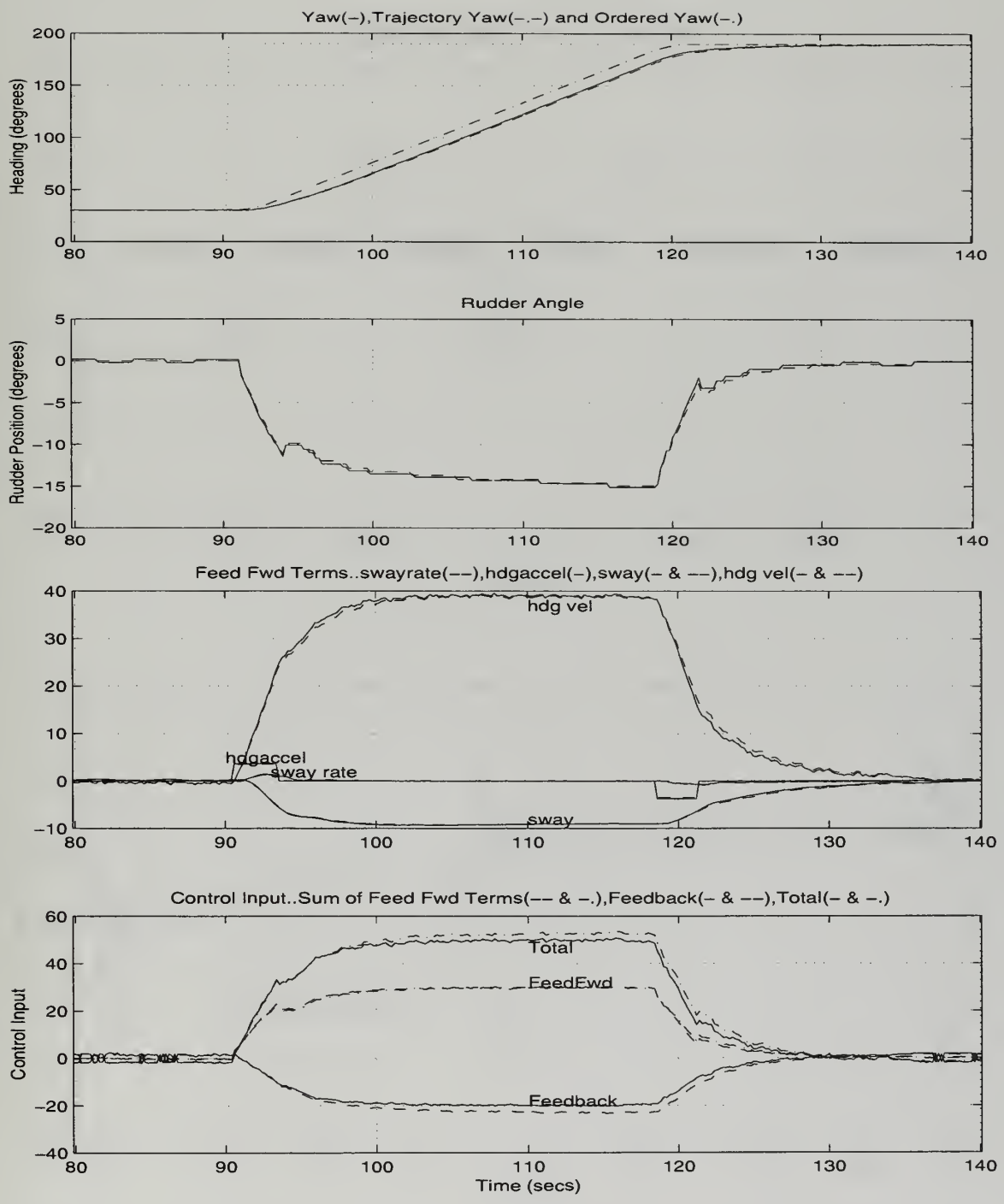

Figure 4.14 Expanded Portion of Figure 4.11 - heading change with control inputs. Small differences between the two plots illustrate that tracking performance can be good even with modeling errors.With the larger modeling errors tracking performance suffers. Vehicle plots with adaptation are shown as solid vice dashed lines.

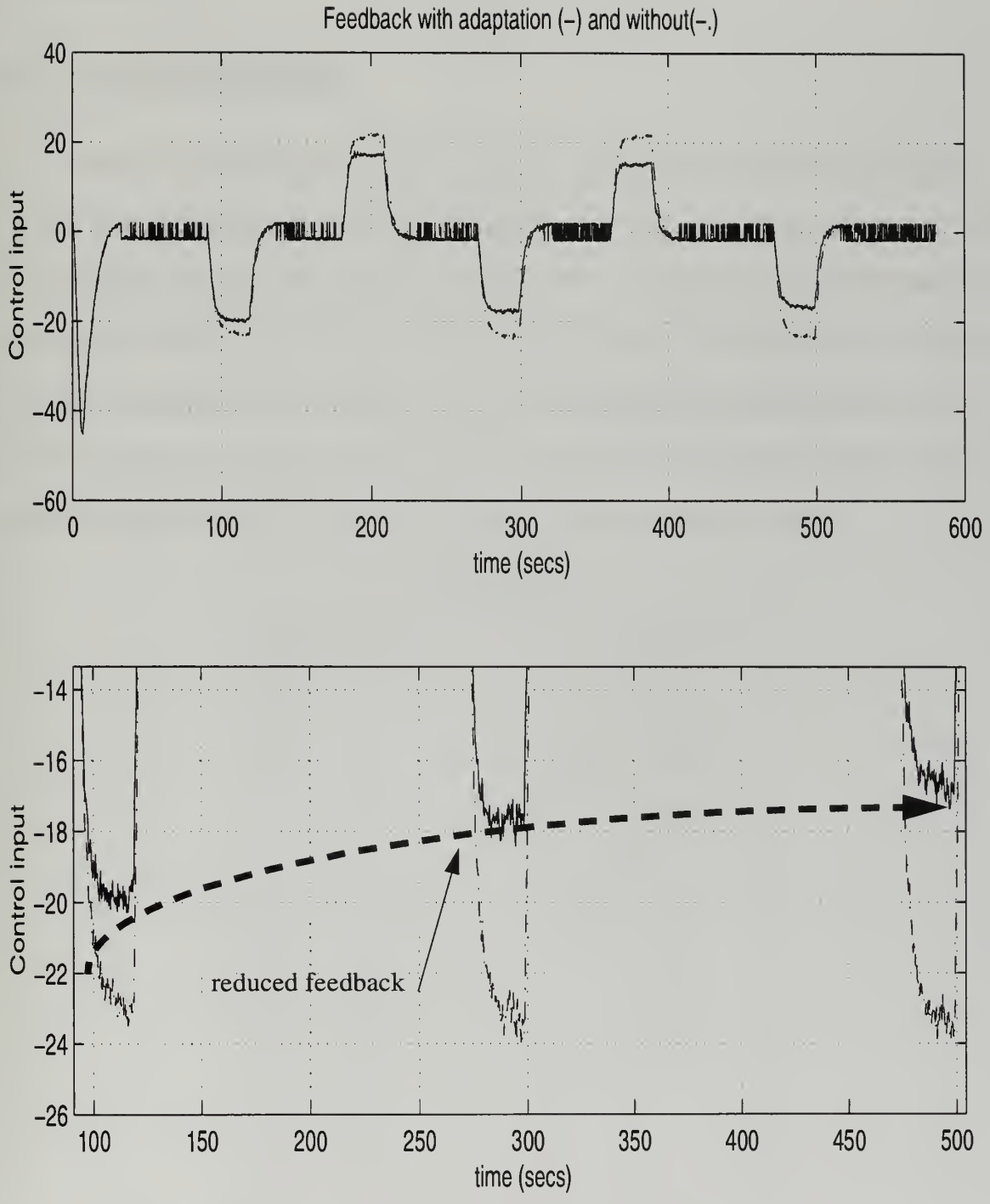

Figure 4.15 Feedback for Heading Changes. Improving estimates of model parameters means that feed forward terms are doing more of the work and less feedback is needed for good tracking performance.

Chapter 5

Directional Stability

Initial testing at Sea Grant Laboratory proves the vehicle is unstable in the yaw plane or that it is directionally controls-fixed unstable. Using Hover's linear model and the hydrodynamic parameters determined at Draper Laboratory[11], the transfer function poles predict vehicle stability in pitch and instability in yaw. The difference between the pitch and yaw planes is the presence of the CG-CB moment arm, Eqs. (5.5) and (5.13), which provides the stability in the pitch plane. The pitch and yaw plane models use the most accurately determined hydrodynamic values to show predicted stability.

Model Parameters			
Maw	396.81 Kg	m	339.26 Kg
Mav	396.81 Kg	xb	.0024 m
Jyy	70.83 Kg	xw	.00243 m
Jamy	30 Kg	xfin	-1.0691 m
Jzz	70.83 Kg	zg	0.082 Kg
Jamz	30 Kg	zb	0.0 m
Zuw	-137.6N	xma	-0.161 m
Zuq	-181.2N	Afin	.0299 m
Muw	-236.Nm	Cl	3.3921 m
Muq	-78.6Nm	B	1173 N
Yuv	-137.6 N	W	1173 N
Yur	-181.2 N		
Nuv	-181.2Nm		
Nur	-78.6 Nm		

Table 5.1: Model Parameters

5.1 Model for Pitch

$$I_\theta \dot{\hat{x}} = A_\theta \hat{x} + B_\theta \hat{u} \tag{5.1}$$

$$\hat{x} = \begin{bmatrix} w & q & \theta \end{bmatrix}^T \tag{5.2}$$

$$u = \begin{bmatrix} \delta e & 1 \end{bmatrix}^T \tag{5.3}$$

$$I_\theta = \begin{bmatrix} m + Maw & -mx_G - x_{ma}Maw & 0 \\ -mx_G - x_{ma}Maw & Jyy + Jamy & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5.4}$$

$$A_\theta = \begin{bmatrix} ZuwU - C_{fin}U & -ZuqU + mU + C_{fin}Ux_{fin} & B-W \\ -MuwU + C_{fin}Ux_{fin} & MuqU - mUx_G - C_{fin}Ux_{fin}^2 & z_B B - z_G W \\ 0 & \cos\phi & 0 \end{bmatrix} \tag{5.5}$$

$$B_\theta = \begin{bmatrix} -C_{fin}U^2 & W-B \\ x_{fin}C_{fin}U^2 & x_B B - x_w W \\ 0 & 0 \end{bmatrix} \tag{5.6}$$

$$C_{fin} = \frac{1}{2}\rho A_{fin} C_l \tag{5.7}$$

$$\theta_{poles} = \begin{bmatrix} -2.0501 \\ -0.10355 + i0.34045 \\ -0.10355 - i0.34045 \end{bmatrix} \tag{5.8}$$

5.2 Model for Yaw

$$I_\psi \dot{\hat{x}} = A_\psi \hat{x} + B_\psi \hat{u} \tag{5.9}$$

$$X_\psi = \begin{bmatrix} v & r & \psi \end{bmatrix} \tag{5.10}$$

$$u = \delta r \tag{5.11}$$

$$I_\psi = \begin{bmatrix} m + Mav & mx_G + x_{ma}Mav & 0 \\ mx_G + x_{ma}Mav & Jzz + Jamz & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5.12}$$

$$A_\psi = \begin{bmatrix} YuvU - C_{fin}U & YurU - mU - C_{fin}Ux_{fin} & 0 \\ NuvU - C_{fin}Ux_{fin} & NurU - mUx_G - C_{fin}Ux_{fin}^2 & 0 \\ 0 & \cos\phi & 0 \end{bmatrix} \tag{5.13}$$

$$B_\psi = \begin{bmatrix} C_{fin}U^2 \\ x_{fin}C_{fin}U^2 \\ 0 \end{bmatrix} \tag{5.14}$$

$$\psi_{poles} = \begin{bmatrix} 0 \\ -2.4441 \\ 0.18688 \end{bmatrix} \tag{5.15}$$

5.3 Hydrodynamic Stability

These poles, Eq. (5.15), show yaw plane open loop instability. Two fundamental ideas are pursued when examining how to reduce power consumption associated with control activity. Chapter 4 discusses the first, the elimination of noise driven actuator motion. The second is the directional stability being determined by the closed loop control system. With constant rudder motion required to keep the bow from being blown off the intended course by any disturbance, an opportunity to reduce the power required for propulsion/

steering purposes exists. If the vehicle controls-fixed stability could be improved, poles in the left half plane or at least close to zero, steering with less rudder activity results. Additionally, with the marginally valid assumption that roll and pitch can be decoupled depending on the trim conditions, less rudder action translates into less rolling. Less rolling means less cross-coupling to pitch and depth control requires less elevator action. Hence, benefits from improving directional stability are gained in both pitch and yaw planes.

In [7], directional stability analysis points out that (5.16) determines stability.

$$C = Y'_v N'_r - N'_v (Y'_r - M') > 0 \tag{5.16}$$

The terms in this expression are non-dimensionalized as indicated.

$$Y'_v = \frac{Y_v}{\frac{\rho}{2} L^2 V} = \frac{Y_{uv} U}{\frac{\rho}{2} L^2 V} = \frac{(-137.6)(1.45)}{\left(\frac{1028}{2}\right) 2.15^2 (1.45)} = -5.79 \cdot 10^{-2}$$

$$Y'_r = \frac{Y_r}{\frac{\rho}{2} L^3 V} = \frac{Y_{ur} U}{\frac{\rho}{2} L^3 V} = \frac{-181.2(1.45)}{\left(\frac{1028}{2}\right) 2.15^3 (1.45)} = -3.55 \cdot 10^{-2}$$

$$N'_v = \frac{N_v}{\frac{\rho}{2} L^3 V} = \frac{N_{uv} U}{\frac{\rho}{2} L^3 V} = \frac{-181.2(1.45)}{\left(\frac{1028}{2}\right) 2.15^3 (1.45)} = -3.55 \cdot 10^{-2}$$

$$N'_r = \frac{N_r}{\frac{\rho}{2} L^4 V} = \frac{N_{ur} U}{\frac{\rho}{2} L^4 V} = \frac{-78.6(1.45)}{\left(\frac{1028}{2}\right) 2.15^4 (1.45)} = -7.16 \cdot 10^{-3}$$

$$M' = \frac{M}{\frac{\rho}{2}L^3} = \frac{-78.6}{\left(\frac{1028}{2}\right)2.15^3} = -7.16 \cdot 10^{-3}$$

So the expression becomes

$$C = (4.15 \cdot 10^{-4}) - (-3.55 \cdot 10^{-2})(-3.55 \cdot 10^{-2} - (-7.16 \cdot 10^{-3}))$$

$$C = -5.91 \cdot 10^{-4}$$

and predicts instability.

For stability, the stern must present more resistance to crossflow and yawing. In the presence of an athwartship velocity, the stern must dominate so the induced torque from crossflow drag swings the bow in the athwartship velocity direction. In the presence of a yaw velocity the stern must again dominate so that the induced torque resists the yawing motion more in the stern than the bow. The method chosen for improving the crossflow drag and its moment arm for the afterbody versus the forebody is adding two fixed vertical fins, one on top and one below, just forward of the rudder. Figure 5.1 shows the hull form with the fixed vertical fins installed.

Adding two fins the same size as the existing rudder surfaces, changes predictions for vehicle directional stability. This prediction is made by adding the fixed vertical fins effect to the hydrodynamic values and determining the transfer function poles. Predicted poles with additional fins are

$$\Psi_{poles} = \begin{bmatrix} 0 \\ -2.7557 \\ -0.0998 \end{bmatrix}$$

With no poles in the right half plane, stability is predicted. Figure 5.2 is a pole plot show-

ing the vertical fins effect. A physical change unmodeled for this study is the docking latch. The latch increases the bow effects and reduces fixed vertical fins effectiveness. Actual results are not expected to be as good as predicted.

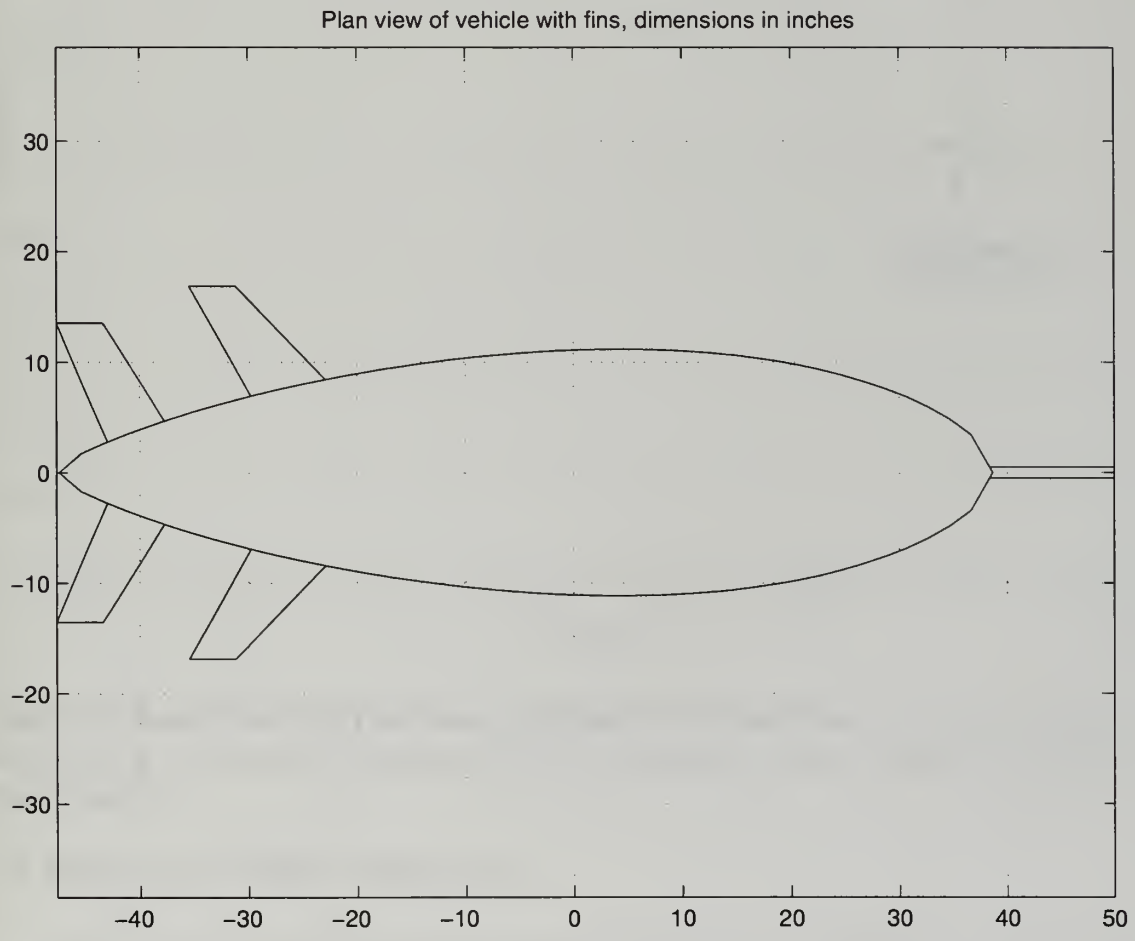

Figure 5.1 Hull Form with Fixed Fins and Latch. Two fixed vertical fins forward of the rudder stabilizes the vehicle in the horizontal plane

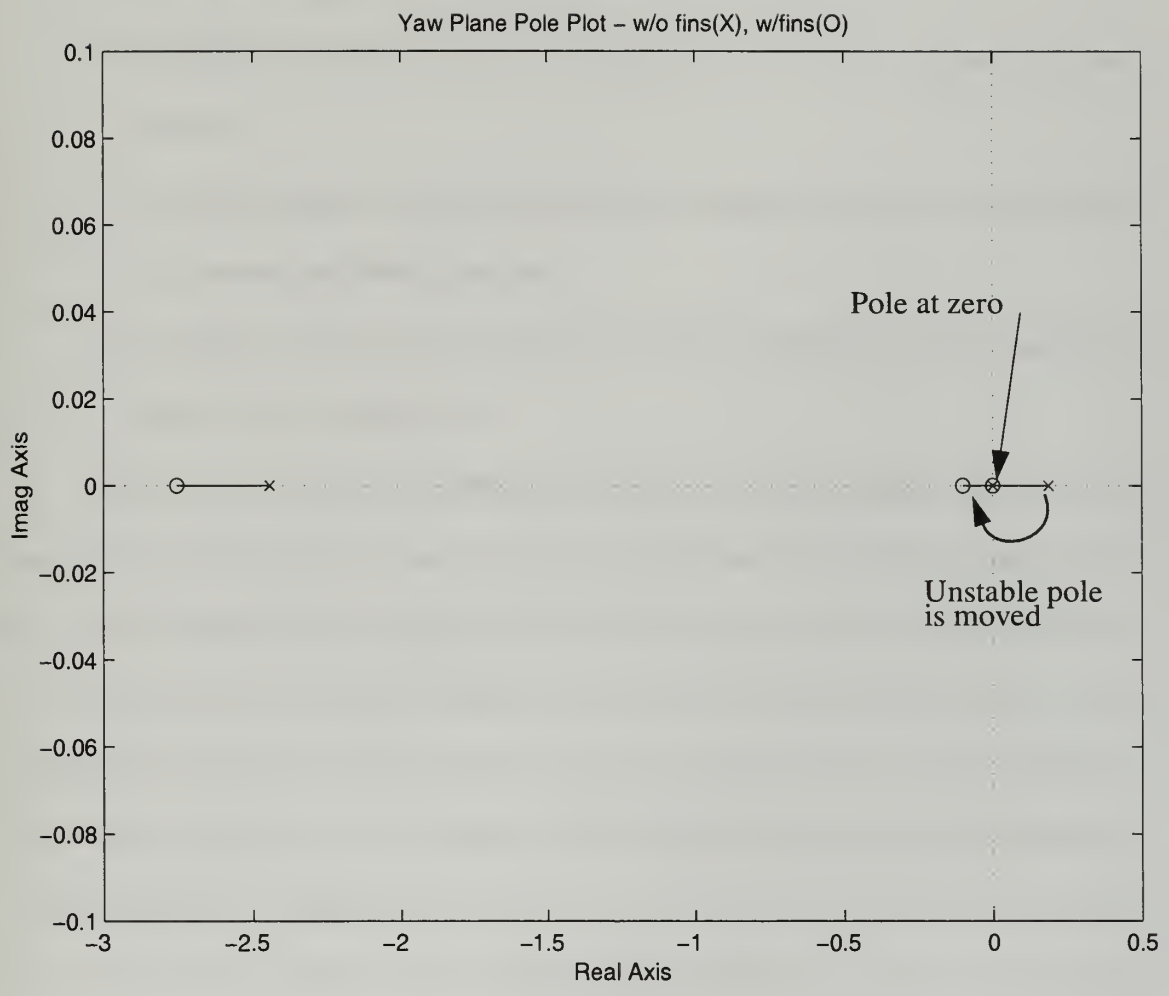

Figure 5.2 Yaw Plane Pole Plot Showing Effect of Vertical Fins. Adding fixed vertical fins moves the one unstable pole into the left half plane predicting controls-fixed directional stability.

5.4 Dieudonne' Spiral Maneuver

With improved stability predictions, the Dieudonne' Spiral Maneuver is chosen to determine the added fins effect on stability. As discussed in [7] this definitive ship trial maneuver identifies the directional stability characteristics. The maneuver is performed by:

1. Steadying on a course and speed for about a minute. Once a steady speed is estab-

lished, no change to thruster rpm is made.

2. The rudder is put over to about 15 degrees and held until a constant yaw rate is achieved.
3. The rudder is then reversed by indexing in 5 degree increments, achieving steady yaw rates at each rudder position.
4. The rudder is reversed again in the same manner, finishing with the rudder at 15 degrees to the original side.

The steady yaw rates as a rudder angle function are then examined. If the yaw rates plot as a single line going from port to starboard rudder and back again, the ship is straight line controls-fixed stable. If the plot forms a hysteresis loop the ship is unstable. The vehicle may turn with the rudder amidships and no external disturbances acting or having acted in the recent past. For the stable ship, only one turning rate for a given rudder position exists, as opposed to the unstable situation where several yaw rates are possible for rudder positions in a neighborhood about zero. In the unstable neighborhood about zero, the vehicle can turn against its rudder. With starboard rudder on, the steady state yaw rate could be to port. Figure 5.3 illustrates a rudder angle-yaw rate plot for a stable and unstable situation. Figure 5.4 shows simulated performance with an updated model to include the vertical stationary fins.

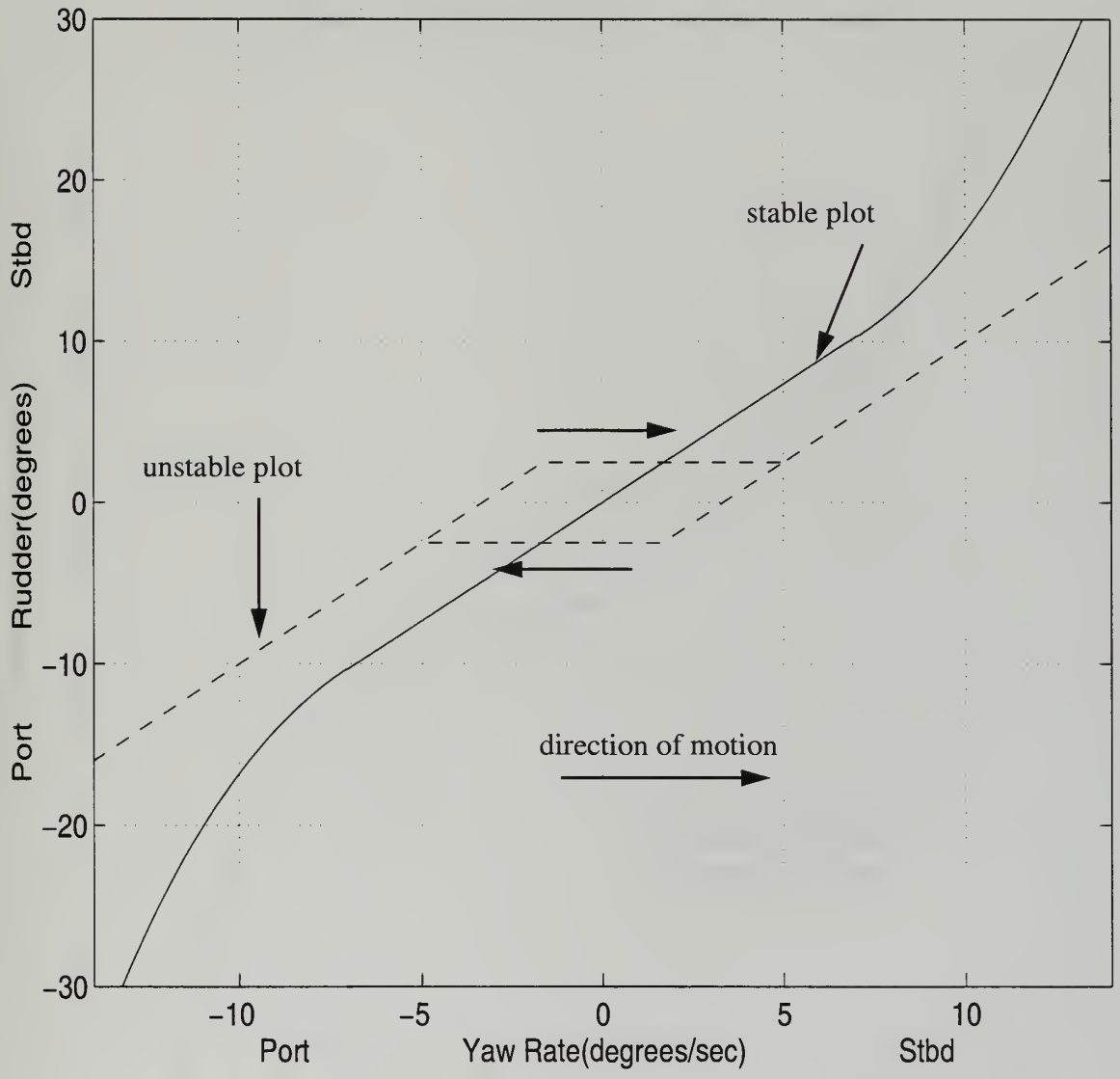

Figure 5.3 Typical Dieudonne' Spiral Maneuver Plot.

For ships that are dynamically and geometrically symmetric about the xz-plane, these rudder curves are symmetric about zero rudder angle and zero yaw rate. An odd number of propellers precludes dynamic symmetry in the xz-plane and displaces the rudder curve to one side or the other depending on screw rotation direction.

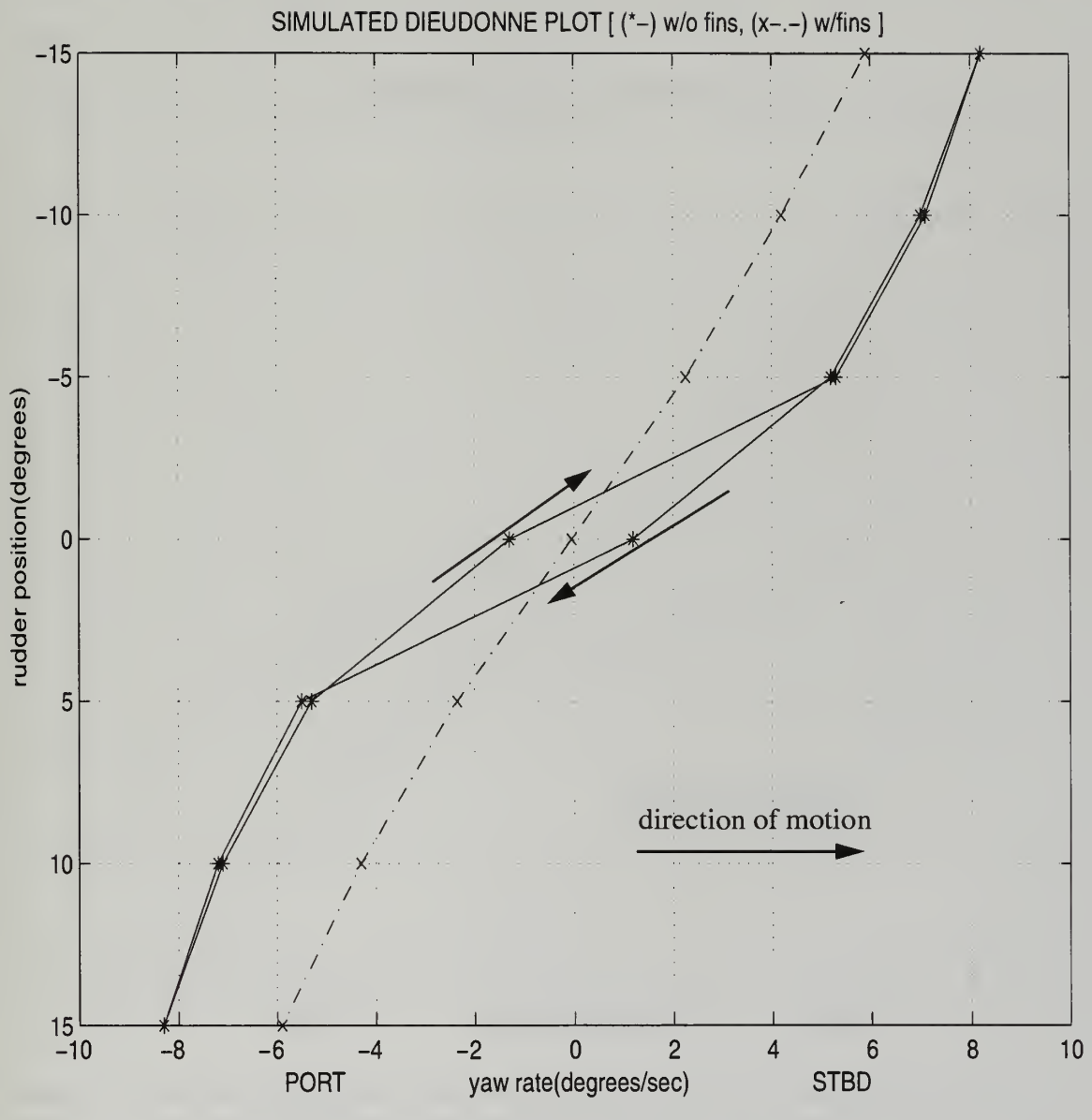

Figure 5.4 Simulated Dieudonne Plot.

Figure 5.5 shows a shifted rudder curve as well as the initial directional instability and the improvement after adding fixed fins. To obtain a closed hysteresis loop for the vehicle with no fins, in Figure 5.5, more data for a larger turning rate to port must be collected. A higher ψ to port would close the hysteresis loop. Although adding fixed fins

improves the directional stability, additional testing is necessary to determine absolute stability.

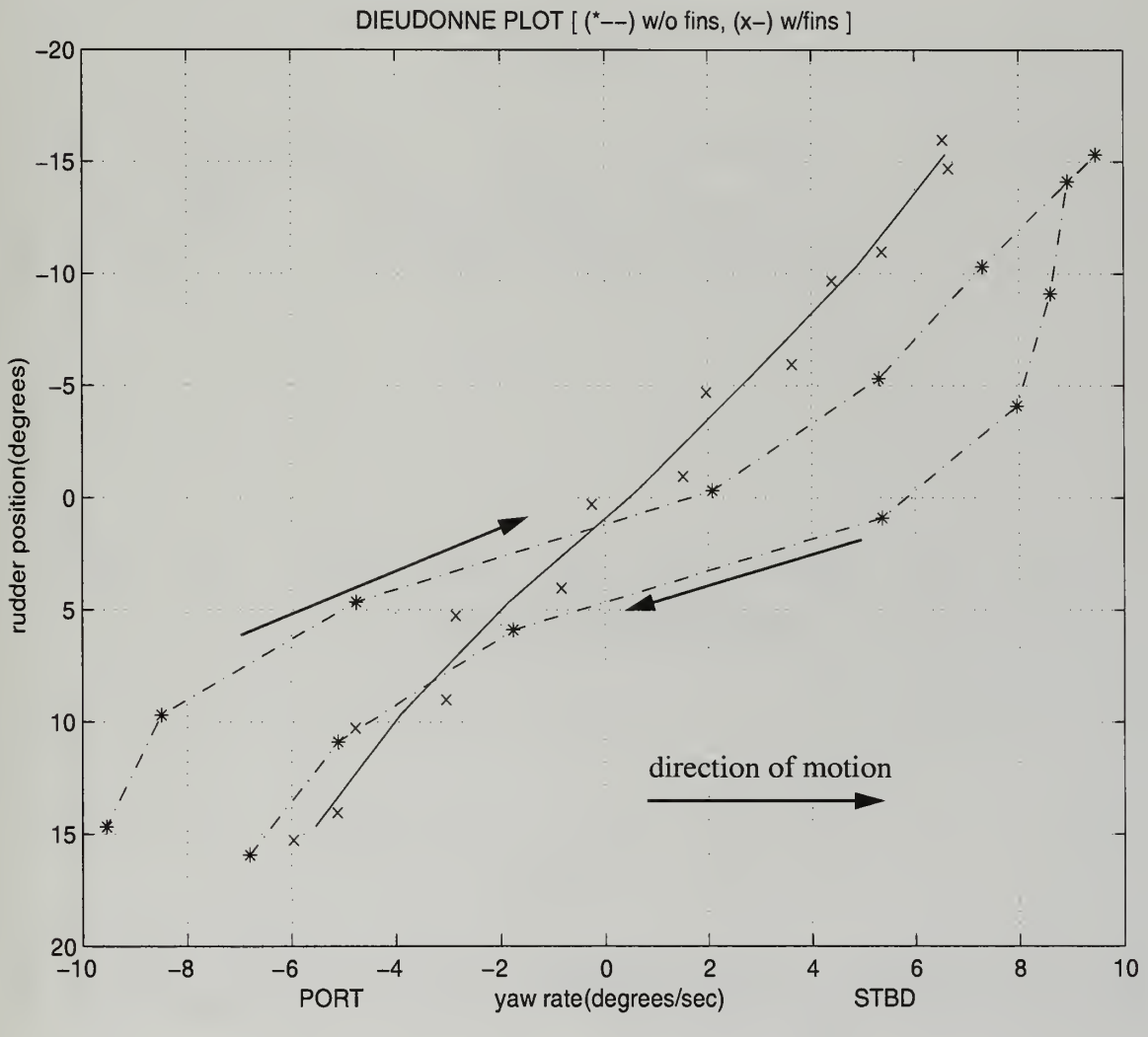

Figure 5.5 Odyssey Dieudonne' Spiral Maneuver Plot. Real vehicle data indicates vertical fins stabilized the vehicle in the horizontal plane. The hysteresis loop is not closed because of the dynamic offset of one propeller. The loop is expected to close at a higher port yaw rate.

Figure 5.6 compares data for vehicle roll and elevator motion during the spiral maneuver. As expected, the rolling and elevator motion with the fixed vertical fins are significantly quieter. The elevator plot shows not only the actuator motion reduction but the

presence of noise in the control input.

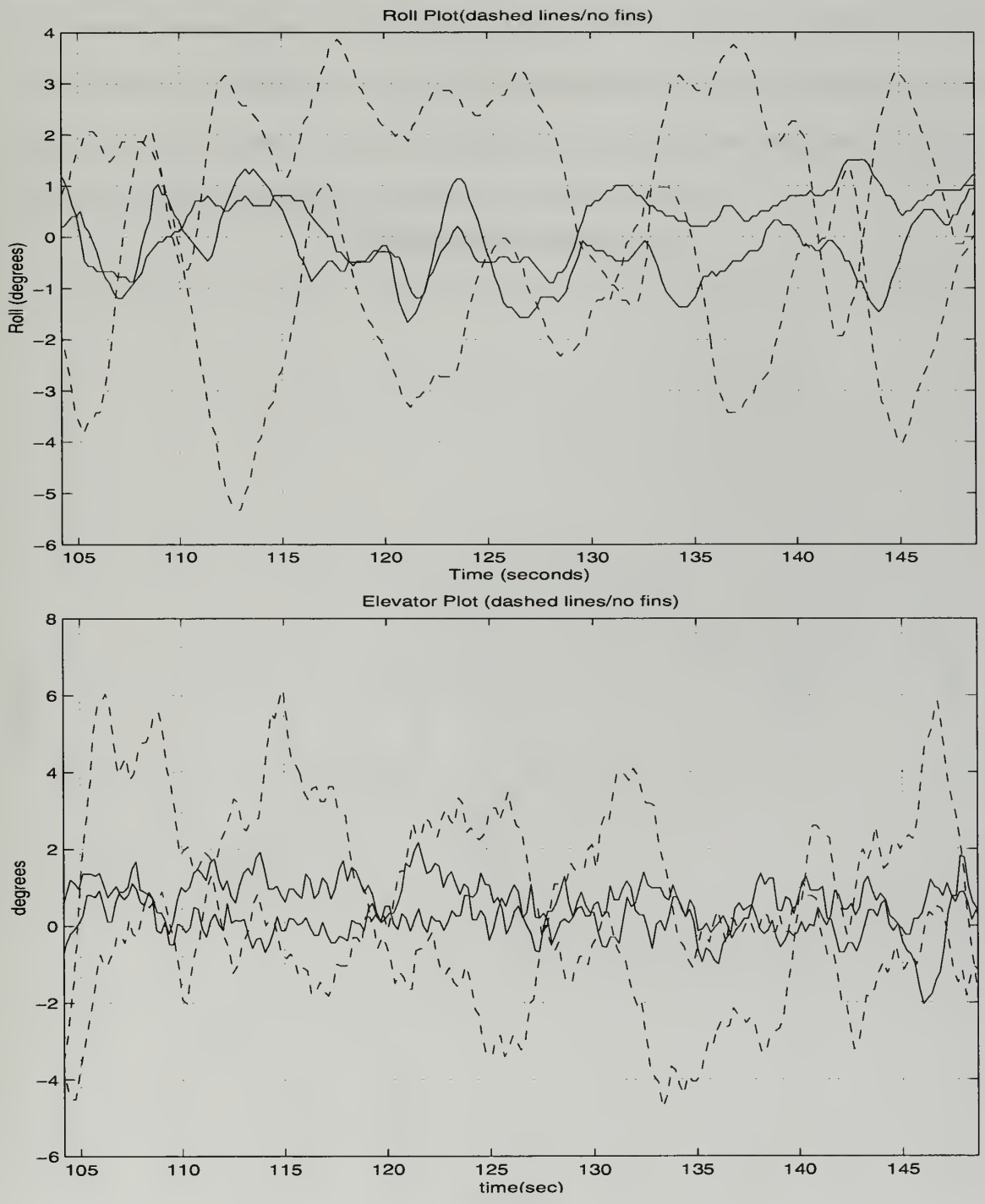

Figure 5.6 Roll and Elevator Motion for Fixed Fin Evaluation. During the spiral maneuver, performance with vertical fins shows quieter vehicle dynamics. Less rolling means rudder effects on pitch are less. As a result less elevator motion is required for depth keeping. Note that with fins installed sensor noise effects are seen in the elevator

motion.

Another characteristic altered by the hull modification is the vehicle rolling into the turn. Without fixed vertical fins, a well defined shift in roll is seen as the vehicle yaw rate changes direction. Figure 5.7 shows the change in roll angle is less pronounced with the fixed fins installed. This indicates a quieting of vehicle dynamics.

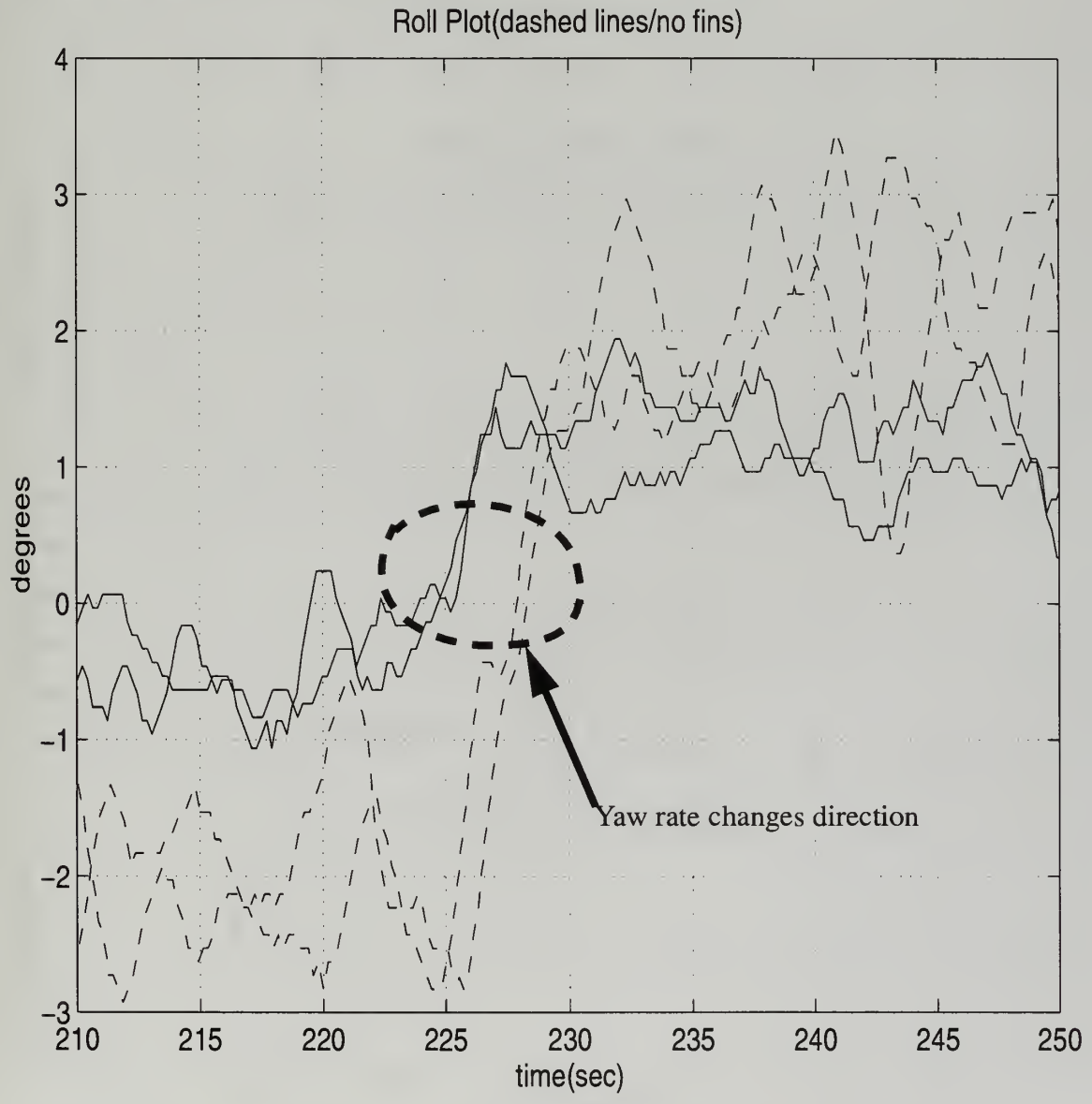

Figure 5.7 Shift in Roll Angle at Zero Yaw Rate. With vertical fins installed the vehicle rolls into the turns less. This plot shows how roll shifts as the turn changes direction. In a homing situation, where the vehicle is hunting back and forth to null a USBL received

bearing, roll magnitude would be reduced.

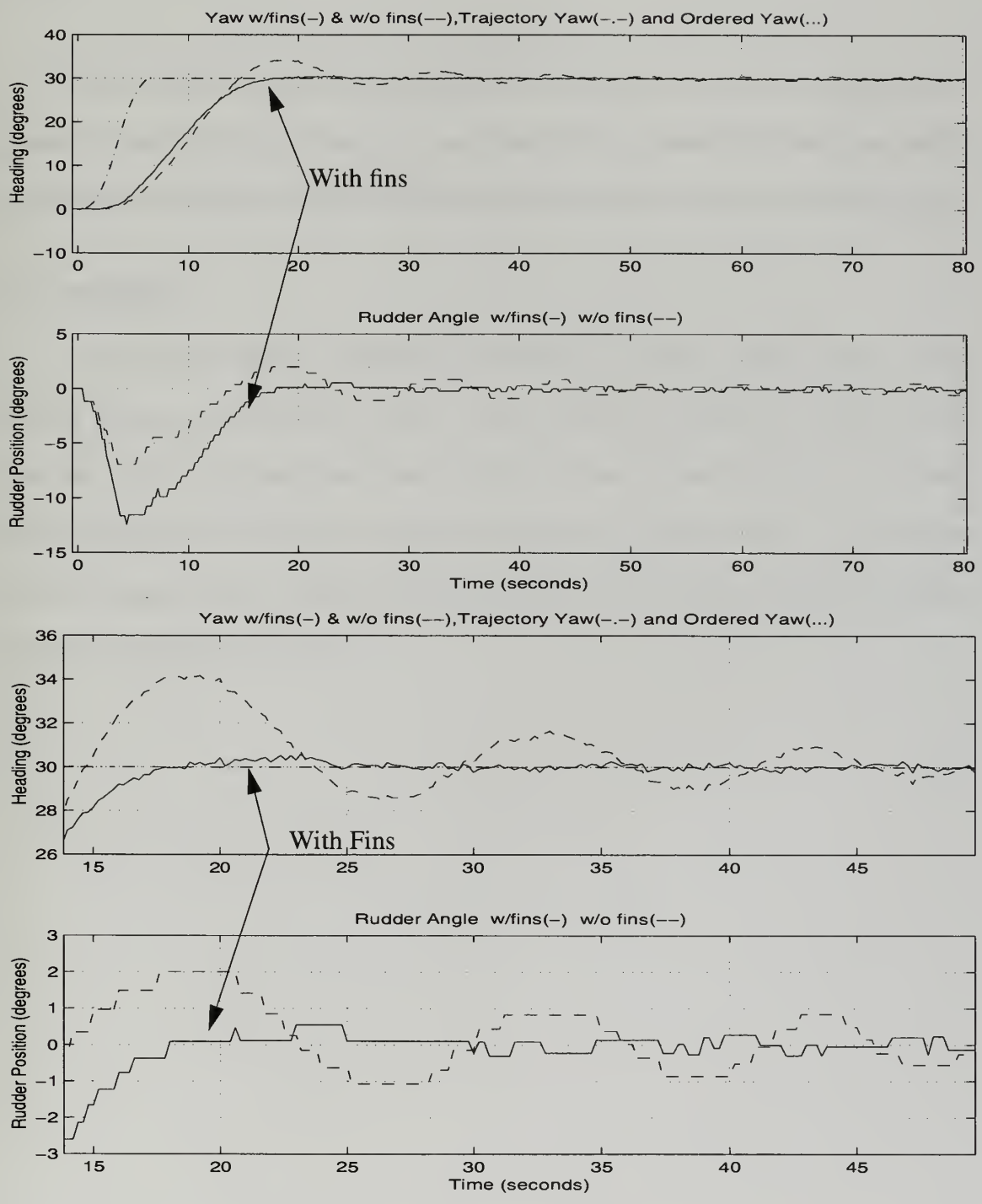

Figure 5.8 Comparison of Simulated Heading Performance with Vertical Fins. Improved performance with vertical fins is demonstrated in this transient. Directional sta-

bility provided by the fixed vertical fins means better performance with less actuator motion.

Figure 5.8 shows a maneuver for a vehicle with and without fins. Here the increased rudder to make the heading change is seen clearly. The expanded plot shows the heading instability for the vehicle without fins and the increased rudder action to maintain heading.

5.5 Summary

Improving the vehicle controls-fixed directional stability reduces control surface activity for both the rudder and elevator. Adding fixed vertical fins to the afterbody stabilizes the vehicle in the horizontal plane. Reduced rudder motion for steering is the direct benefit. Less obvious benefits are reduced rolling and pitching leading to less elevator motion for depth control. Quieter vehicle dynamics means energy savings from reduced fin drag and actuator motion.

Chapter 6

Conclusions and Future Work

Improvements in the three areas examined achieved marked improvement in vehicle performance. Analysis of each change shows that the modifications made as a whole, or selective changes made incrementally, provide better vehicle performance with less energy expended. This section presents conclusions concerning each modification. Additionally the proposed modifications are ordered in easiest to most involved as far as implementation difficulty.

6.1 Actuator Filtering

A rudimentary filter based on the physical gear backlash limitation eliminates excessive actuator motion driven by sensor noise. With little or no noticeable effect on performance, the filter reduced actuator motion from 5 Hz to less than 1 Hz. This analysis uses simulations, but initial actuator motion evaluation, gear backlash and sensor noise levels indicates similar results are possible with the vehicle.

Filter implementation is by far the easiest as it requires a software change with evaluations having minimal impact on planned operations. Determining the optimum filter settings can be done gradually and evaluated quickly during the normal course of any vehicle deployment preparations. Decreasing actuator motion from 5 Hz to less than 1 Hz greatly reduces the power consumed for vehicle control. Working amp-hour sensors allow either a one time power budget analysis or continual evaluation. Benefits from each modification presented here could be quantified and further evaluated.

6.2 Directional Stability

Starting with an initial controls-fixed directional instability, adding two fixed vertical stationary fins achieves stable open-loop dynamics. These fins greatly reduce actuator motion necessary to maintain a steady course. With inherent rolling of the vehicle, rudder cross coupling causes pitching that can not be eliminated, only reduced. Less rudder motion results in less pitching and rolling and a corresponding reduction in elevator motion. Comparing the costs and benefits, the drag reduction from less actuator motion far outweighs any increased drag from the addition of fixed fins. Less control surface drag, together with the power savings from reduced actuator motion in both horizontal and vertical planes achieves even larger reductions in power consumption by propulsion and vehicle control.

This modification requires approximately $150 of material for the two fins and fin boxes for mounting. Fin installation requires minor machining and hull modifications that could be accomplished for one vehicle in eight man-hours. Per vehicle man-hour costs are reduced if making multiple vehicle modifications. There are no added operational requirements for evaluations. Improvements obtained in vehicle control alone make this modification worthwhile. Once made, improvements in vehicle performance are likely during homing as well. With less pitching and rolling during homing, less data scatter from the Ultra Short Base Line Navigation system and a corresponding improvement in homing performance is probable. Other imaging packages such as side scan sonar benefit greatly from a stable platform. Reduced vehicle motion provides side scan sonar images with greater clarity and higher resolution.

6.3 Controller

An adaptive model based sliding controller designed and implemented in simulation shows robustness to speed changes and modeling errors. With model parameter adaptation, an already high performance level improves. The ability to enjoy consistent performance without expending significant man-hours following minor instrumentation changes is certain.

Sliding adaptive controller implementation requires software integration which should be a minimal effort as the controller has been kept current with software revisions. Once installed, initial controller tuning is estimated to take a couple days of vehicle operations. This initial controller tuning is expected to be a one time cost as subsequent operations will be free of this effort. Adaptation can be turned on and off as desired. Small incremental changes in the model parameters can improve performance without impacting vehicle operations. After an adequate evaluation period following configuration changes, new model parameters can be fixed and adaptation disabled if desired.

The changes in operational scenarios possible with the sliding adaptive controller are enormous. Operating at a speed that maximizes propulsion efficiency during transits and then changing speed to optimize sensor performance during sampling can be a routine matter. Other operational possibilities are:

-maintaining a constant speed over ground in changing current conditions during sidescan sonar imaging. Constant image resolution could be maintained.

- "Soft Docking" or slowing the vehicle during a homing run to minimize impact velocity. This reduces mechanical shock to electronic equipment inside the spheres as well as the chance of latch component failure from repeated docking impacts in

the colder ocean waters. Improved final homing maneuver effectiveness means fewer "misses".

- Increased maneuverability during bottom following and obstacle avoidance behaviors.

There are bound to be many other changes in operational scenarios and vehicle behaviors possible once operators realize that constant speed missions are no longer a restriction.

After the Sontek Acoustic Doppler Velocimeter installation, the computation intensive estimator can be eliminated. Accurate multi-axis velocities from the ADV void the necessity for state estimation. This improvement will be practical if the sensor accuracy is high and noise level sufficiently low.

Appendix

Controller Equations

A.1 Parameter Initialization

```
#define D2R(d) (d*PI/180.0)
#define R2D(r) (r*180.0/PI)
double GAMMA[11]={5e-2,3e-1,5e-2,5e-2,5e-2,1e-1,1e-2,2e-1,5e-3,1e-1,1e-2};
static double Ahatdot[11]={0,0,0,0,0, 0,0,0,0,0};

/*THIS IS GOOD FOR NO FIN VEHICLE*/
static double Ahat[11]={-6.3062e+01,1.0083e+02,1.2563e+02,1.3884e+02,
0,6.3062e+01, 1.0083e+02 ,-1.8133e+02, 7.9336e+01, 9.6186e+01,1.0 };

/*THIS IS GOOD FOR FINNED VEHICLE*/
 /*static double Ahat[11]={-1.0299e+02,1.0083e+02,1.4357e+01,2.5775e+02,
0,6.3062e+01, 1.0083e+02 ,-1.8133e+02, 7.9336e+01, 9.6186e+01,1.0 };    */

/*****************************************************************/
/* THIS PROGRAM WORKS WITH DEGREES, NOT RADIANS . VALUES FOR
ANGLES WITH A FEW EXCEPTIONS ARE STORED IN THE DATA STRUCTURE
AS RADIANS BUT ARE CONVERTED TO DEGREES FOR MOST CALCULA-
TIONS. FOR THE INPUT CALCS, TERMS WITH DEGREES HAVE TO BE CON-
VERTED BACK TO RADIANS FOR CONSISTENT UNITS     */
/*****************************************************************/
 /* load present vehicle state  */
 heading_rate =R2D( sensor(M_YAW_RATE) );   /* IN DEGREES/SEC */
 des_speed = vvp->c[0].speed[0];/* IN METERS/SEC */
 speed = sensor(E_SPEED);
 pitch_limit=26*speed-9.0;
 pitch_limit_traj=26*des_speed-9.0;
 maxdr = sin(D2R(pitch_limit_traj))*des_speed ;/* IN METERS/SEC */
 pitch_rate = R2D(sensor(M_PITCH_RATE) );   /* IN DEGREES/SEC */
 depth = sensor(M_DEPTH); /* IN METERS */
 depth_rate = (depth-old_depth)/dt;   /* IN METERS/SEC */
 c_depth =  sensor(C_DEPTH);          /* IN METERS */
 sway = sensor(E_V);/* IN METERS/SEC */
 sway_rate = sensor(E_VD);  /* IN METERS/SEC^2 */
 heave = sensor(E_W);/* IN METERS/SEC */
 heave_rate = sensor(E_WD);  /* IN METERS/SEC^2 */
 roll = R2D(sensor(M_ROLL) );        /* IN DEGREES */
```

```
pitch = R2D(sensor(M_PITCH));              /* IN DEGREES */
heading = R2D(sensor(M_YAW));
c_heading  =R2D( sensor(C_HEADING));       /* IN DEGREES */
lambda_theta = sensor( LAMBDA_THETA );
k2_p = sensor( K2_P);
lambda_p = sensor( LAMBDA_P );
lambda_z = sensor( LAMBDA_Z );
lambda_h = sensor( LAMBDA_H );
k1_h =sensor( K1_H );
maxhr = R2D(des_speed *sensor(UO_MAX_HR)); /* IN METERS/SEC */
amax_depth=.01*des_speed ; /* IN METERS/SEC^2 */
amax_hdg =2*des_speed ;/* IN DEGREES/SEC^2 */
rtime = sensor( M_PRESENT_TIME );
dt = rtime - last_time;
if (dt > .3) dt = .3 ;
  if (rtime <= 2.*dt){
  ctr=0;
  while (ctr <=9){
    Ahat_max[ctr] = Ahat[ctr]*1.2;
    Ahat_min[ctr] = Ahat[ctr]*.8;
    ctr +=1;
    }

init_traj_z(depth,c_depth,amax_depth, maxdr,rtime,&depth_traj);
depth_traj.x = depth ;
depth_accel=0;
depth_rate=0;
      }

 if (rtime <= 2.*dt){
init_traj_hdg(heading,c_heading ,amax_hdg, maxhr,rtime,&hdg_traj) ;
hdg_traj.x = heading ;
}
```

A.2 Pitch/Depth Control

```
/*-------------------------------------------------------------------------
|
|Adaptive controller for Depth control
|
|Phil LeBas 7/96
|
|-------------------------------------------------------------------------*/
```

```
/* added  calculation for the adaptive controller */

 /*--- Do dynamic control of depth....                */


/* check to see if the commanded depth has been changed
- if so compute the new trajectory */

if (old_c_depth != c_depth) init_traj_z(depth,c_depth,amax_depth,
maxdr,rtime,&depth_traj);

/*************************************************************/
/* THIS TRAJECTORY GENERATOR WORKS WITH DEGREES, NOT RADIANS */
/*************************************************************/
do_traj_z(&depth_traj, rtime,&px, &pxdot);
des_depth = depth_traj.x;
if (depth_traj.phase == TR_ACCELERATING) ACCEL =1;
if (depth_traj.phase == TR_DECELERATING) ACCEL =-1;
if (depth_traj.phase == TR_CONSTANT_VELOCITY || depth_traj.phase ==TR_DONE)
ACCEL =0;
if (speed < 0.5)
   speed = 0.5;

des_depth_rate = depth_traj.xdot;
depth_accel= (depth_rate - old_depth_rate)/dt;
heave_accel= (heave_rate - old_heave_rate)/dt;
des_depth_accel = ACCEL*depth_traj.amax*sgn(des_depth_rate);
des_heave_accel = 0.0;
/************************************/
z_tilde = depth - des_depth;
z_tilde_dot = depth_rate - des_depth_rate;
s_z = z_tilde_dot + lambda_z * z_tilde;
s_delta_z= (s_z-phi_z*sat(s_z/phi_z));
/************************************/
des_pitch_rate  = -R2D(((des_depth_accel + heave_rate*cos(D2R(pitch)))
/ (heave*sin(D2R(pitch))+ speed*cos(D2R(pitch))))
-k2_p/2.5*s_z);/* IN DEGREES/SEC  */
des_pitch_rate = limit( des_pitch_rate,maxhr,-maxhr);
pitch_accel= (pitch_rate - old_pitch_rate)/dt;/* IN DEGREES/SEC^2 */
des_pitch_accel =(des_pitch_rate - old_des_pitch_rate)/dt; /* IN DEGREES/SEC^2 */
des_pitch_accel = limit( des_pitch_accel,amax_hdg,-amax_hdg);
des_pitch_int += des_pitch_rate*dt;
des_pitch = limit( des_pitch_int, pitch_limit, -pitch_limit);
```

```
 /* set up basic parameters */
x_tilde = depth- des_depth;
W_tilde = heave_rate - des_heave_rate;
theta_tilde = pitch - des_pitch;                    /* IN DEGREES  */
q_tilde = pitch_rate - des_pitch_rate;/* IN DEGREES/SEC  */
ref_heave_rate = des_heave_rate - lambda_p * x_tilde;
ref_pitch_vel = des_pitch_rate - lambda_p * theta_tilde;/* IN DEGREES/SEC  */
ref_pitch_accel = des_pitch_accel - lambda_p*q_tilde;/* IN DEGREES/SEC^2  */
ref_heave_accel = des_heave_accel - lambda_p*W_tilde;
s2= D2R(q_tilde + lambda_theta * theta_tilde);/* IN RADIANS/SEC  */
s_delta_2= (s2-phi*sat(s2/phi));/* IN RADIANS/SEC */

if (rtime <= 2.*dt){
B22 =-( sensor( REAL_VEH_W )*Xg - sensor( REAL_VEH_B )*Xb);
 }
 else {
B22=- Ahat[10];
 }

/*   calculate the new guesstimates on the system parameters */
if (fabs(des_speed-speed)<=0.2*des_speed && rtime >= 60){
Ahatdot[5]=-GAMMA[5]*s_delta_2*heave_rate ;
Ahatdot[6]=-GAMMA[6]*s_delta_2*ref_pitch_accel ;
Ahatdot[7]=-GAMMA[7]*s_delta_2*speed*heave;
Ahatdot[8]=-GAMMA[8]*s_delta_2*speed* ref_pitch_vel;
Ahatdot[9]=-GAMMA[9]*s_delta_2*sin(D2R(pitch));
Ahatdot[10]=-GAMMA[10]*s_delta_2*cos(D2R(pitch));

Ahat[5]+= Ahatdot[5]*dt;
Ahat[6]+= Ahatdot[6]*dt;
Ahat[7]+= Ahatdot[7]*dt;
Ahat[8]+= Ahatdot[8]*dt;
Ahat[9]+= Ahatdot[9]*dt;
Ahat[10]+= Ahatdot[10]*dt;

Ahat[5] = limit( Ahat[5], Ahat_max[5],Ahat_min[5]);
Ahat[6] = limit( Ahat[6], Ahat_max[6],Ahat_min[6]);
Ahat[7] = limit( Ahat[7], Ahat_max[7],Ahat_min[7]);
Ahat[8] = limit( Ahat[8], Ahat_max[8],Ahat_min[8]);
Ahat[9] = limit( Ahat[9], Ahat_max[9],Ahat_min[9]);
Ahat[10] = limit( Ahat[10],  -B22*1.5,-B22*-1.5);
}
/*****************************************************************/
```

```
/* THIS PROGRAM WORKS WITH DEGREES, NOT RADIANS . VALUES FOR
ANGLES WITH A FEW EXCEPTIONS ARE STORED IN THE DATA STRUCTURE
AS RADIANS BUT ARE CONVERTED TO DEGREES FOR MOST CALCULA-
TIONS. FOR THE INPUT CALCS, TERMS WITH DEGREES HAVE TO BE CON-
VERTED BACK TO RADIANS FOR CONSISTENT UNITS     */
/*****************************************************************/
M_elev_hat = speed*(Ahat[7]*heave + Ahat[8]* D2R(ref_pitch_vel)) +
Ahat[9]*sin(D2R(des_pitch)) +Ahat[5]*heave_rate + Ahat[6]*D2R(ref_pitch_accel) +
Ahat[10]*cos(D2R(des_pitch));

M_elev = M_elev_hat-k2_p*s2;
/* This keeps the control surfaces from going crazy during speed changes */
if (speed <= des_speed)
    control_surface_k=(des_speed*des_speed*1025*3.3921* .0299);
else
   control_surface_k=(speed*speed*1025*3.3921* .0299);

Del24=M_elev/(-1.0691*control_surface_k);

Del24 = limit( Del24, MAX_CONTROL, -MAX_CONTROL);

/*****************************************************************/
/* THIS PROGRAM WORKS WITH DEGREES, NOT RADIANS . VALUES FOR
ANGLES WITH A FEW EXCEPTIONS ARE STORED IN THE DATA STRUCTURE
AS RADIANS BUT ARE CONVERTED TO DEGREES FOR MOST CALCULA-
TIONS. FOR THE INPUT CALCS, TERMS WITH DEGREES HAVE TO BE CON-
VERTED BACK TO RADIANS FOR CONSISTENT UNITS     */
/*****************************************************************/

/* If the change in actuator position isn't greater
 than 1/2 degree it isn't worth it. */

    if (fabs(sensor(C_ELEVATOR_ANGLE)-Del24)>=.5*(PI/180)*(1.0/speed) )
    {
    update( C_ELEVATOR_ANGLE, Del24);  /* positive elevator angle > dive */
    }
    else
    {
    update( C_ELEVATOR_ANGLE, sensor(C_ELEVATOR_ANGLE));
    }

 break;
```

A.3 Heading Control

```
/*-------------------------------------------------------------------------
|
|Adaptive controller for Heading control
|
|Phil LeBas 10/96
|
|-------------------------------------------------------------------------*/
if (old_c_heading != c_heading) init_traj_hdg(heading,c_heading ,amax_hdg,
maxhr,rtime,&hdg_traj) ;
do_traj_hdg(&hdg_traj, rtime,&px, &pxdot);

/*****************************************************************/
/* THIS PROGRAM WORKS WITH DEGREES, NOT RADIANS . VALUES FOR
ANGLES WITH A FEW EXCEPTIONS ARE STORED IN THE DATA STRUCTURE
AS RADIANS BUT ARE CONVERTED TO DEGREES FOR MOST CALCULA-
TIONS. FOR THE INPUT CALCS, TERMS WITH DEGREES HAVE TO BE CON-
VERTED BACK TO RADIANS FOR CONSISTENT UNITS     */
/*****************************************************************/
/* added  calculation for the adaptive controller */
des_head = hdg_traj.x;/* IN DEGREES */
des_heading_rate = hdg_traj.xdot;
if (hdg_traj.phase == TR_ACCELERATING) accel =1;
if (hdg_traj.phase == TR_DECELERATING) accel =-1;
if (hdg_traj.phase == TR_CONSTANT_VELOCITY || hdg_traj.phase ==TR_DONE)
accel =0;

/*  This finds the shortest direction for the turn  */
deltaheading = hdg_traj.xend - hdg_traj.xstart;
while(deltaheading > 180)deltaheading -= 360 ;
while(deltaheading < -180)deltaheading += 360 ;

des_head_accel = sgn(deltaheading)*accel*hdg_traj.amax;
head_accel = (heading_rate - old_heading_rate)/dt;
yaw_tilde = mod_180(heading - des_head);
r_tilde = heading_rate - des_heading_rate;
s1= D2R(r_tilde + lambda_h * yaw_tilde);
s_delta_1= s1-PHI*sat(s1/PHI) ;
des_sway = 0;
des_sway_rate =0;
v_tilde = sway - des_sway;
ref_head_vel = des_heading_rate - lambda_h * yaw_tilde;
ref_hdg_accel= des_head_accel;/* - lambda_h * r_tilde;*/
```

```
ref_sway = des_sway ;
ref_sway_rate = des_sway_rate - lambda_h * v_tilde;

/*   calculate the new guesstimates on the system parameters */
Ahatdot[0]=-GAMMA[0]*s_delta_1*sway_rate;
Ahatdot[1]=-GAMMA[1]*s_delta_1* ref_hdg_accel;
Ahatdot[2]=-GAMMA[2]*s_delta_1* speed*sway;
Ahatdot[3]=-GAMMA[3]*s_delta_1* speed*ref_head_vel;

Ahat[0]+= Ahatdot[0]*dt;
Ahat[1]+= Ahatdot[1]*dt;
Ahat[2]+= Ahatdot[2]*dt;
Ahat[3]+= Ahatdot[3]*dt;
Ahat[4]+= Ahatdot[4]*dt;

Ahat[0] = limit( Ahat[0], Ahat_max[0], Ahat_min[0]);
Ahat[1] = limit( Ahat[1], Ahat_max[1], Ahat_min[1]);
Ahat[2] = limit( Ahat[2], Ahat_max[2], Ahat_min[2]);
Ahat[3] = limit( Ahat[3], Ahat_max[3], Ahat_min[3]);
Ahat[4] = limit( Ahat[4], Ahat_max[4], Ahat_min[4]);

M_rudder_hat = speed*(Ahat[2]*sway + Ahat[3]*D2R( ref_head_vel))
+Ahat[0]*sway_rate + Ahat[1]*D2R(ref_hdg_accel);

M_rudder = M_rudder_hat - k1_h*s1;
/*control_surface_k=(des_speed*des_speed*.5*1025*3.39* .0299);*/
Del13=M_rudder/(-1.0691*control_surface_k);
Del13 = limit( Del13, MAX_CONTROL, -MAX_CONTROL);

/* If the change in actuator position isn't greater
 than 1/2 degree it isn't worth it. */
    if (fabs(sensor(C_RUDDER_ANGLE)-Del13) >= .5*(PI/180)*(1.0/speed) )
    {
    update( C_RUDDER_ANGLE, Del13);
    }
    else
    {
    update( C_RUDDER_ANGLE, sensor(C_RUDDER_ANGLE));
 }
    /*Positive rudder means turn to left/port*/
}
 /*-----------------------------------------------------------------
     Store some vehicle state values for next iteration....
```

```
   ----------------------------------------------------------------*/
   old_des_head = des_head;
   old_des_heading_rate = des_heading_rate;
   old_heading_rate  = heading_rate;
   old_depth = depth;
   old_c_depth = c_depth;
   old_depth_rate = depth_rate;
   old_heave_rate = heave_rate;

   old_des_depth = des_depth;
   old_des_depth_rate = des_depth_rate;
   old_c_heading= c_heading ;
   old_head = heading;
   old_pitch = pitch;
   old_pitch_rate = pitch_rate;
   old_des_pitch = des_pitch;
   old_des_pitch_rate = des_pitch_rate;
   old_speed = speed;
   last_time = rtime;
}
```

A.4Heading and Pitch/Depth Control Functions

```
/* keeps heading between 0-2PI, only corrects once though */
void wrap_heading(double *h)
{
       while(*h > TWOPI)*h -= TWOPI ;
       while(*h < 0)*h += TWOPI ;
}
/*================================================*/
/* corrects the difference between 2 headings to keep the value between
+- pi.  In other words, it takes the shortest distance around the circle
*/
double delta_heading(double dh)
{
      double result ;
      result = dh ;
      while(result > PI)result -= TWOPI ;
      while(result < -PI)result += TWOPI ;
      return(result) ;
}
/*================================================*/
double delta_heading_deg(double dh)
```

```
{
      double result ;
      result = dh ;
      while(result > 180)result -= 360 ;
      while(result < -180)result += 360 ;
      return(result) ;
}
/*=================================================*/
double dista(double accel, double t)
{
return(.5*accel*t*t);
}
 /*============================================================
|   LIMIT:
|     Makes sure a variable falls within defined limits...
| ============================================================ */
double limit( double value, double max, double min )
{
   double dummy;

   if ( max < min )
   {
      dummy = max;
      max = min;
      min = dummy;
   }

   if (value > max)
      value = max;
   if (value < min)
      value = min;
   return value;
}
/*=================================================*/
static double sgn( double value )
{
   double sign;

   if (value < 0.0)
sign = -1.0;
   if (value >= 0.0)
sign = 1.0;
   return sign;
```

```
}
/*================================================*/
static double sat(double x)
{
   double y;
   if ( fabs(x) > 1. )
y = sgn(x);
   else
y = x;
   return y;
}
/*================================================*/
static double mod_180(double angle)
{
if(fabs(angle)>360.0)
  angle = angle - (double)((int)(angle/360.0))*360.0;
if(angle>=180.0)
  angle = angle - 360.0;
if(angle<-180.0)
  angle = angle + 360.0;
return(angle);
}
```

A.5 Trajectory Generator

```
/*================================================*/
/* initialize the trajectory structure
you also need to set the trajectory displacement (.x) to the
current hdg state to initialize*/
void init_traj_hdg(double xstart, double xend, double amax, double vmax, double t,
traj_new *tr)
{
double taccel, dx, xtriangle, xconst_vel ;

tr->tstart = t ;
tr->xstart = xstart ;
 tr->xend = xend ;
tr->amax = fabs(amax) ;
tr->vmax = fabs(vmax) ;
tr->x = xstart ;
tr->xdot = 0.0;
tr->phase = TR_ACCELERATING;
taccel = tr->vmax/tr->amax;
```

```
/* compute the distance traveled if we accelerate to max speed and
then decelerate back to zero speed */
xtriangle = tr->amax*taccel*taccel ;
dx = delta_heading_deg(xend-xstart) ;

/* test if the needed velocity profile is a trapezoid */
if(fabs(dx) > xtriangle)
{
tr->t1 = tr->tstart + taccel ;
xconst_vel = fabs(dx)-xtriangle ;   /* distance covered at constant vel*/
tr->t2 = tr->t1 + xconst_vel/tr->vmax ;
tr->tend = tr->t2 + taccel ;
tr->type = TR_TRAPEZOIDAL ;
}
else
{
 taccel = sqrt(fabs(dx)/tr->amax) ;
tr->t1=tr->t2 = tr->tstart + taccel ;
tr->tend = tr->t1+taccel ;
tr->type = TR_TRIANGULAR ;
}
}

/*================================================*/
void init_traj_z(double xstart, double xend, double amax, double vmax, double t, traj_new
*tr)
{
double taccel, dx, xtriangle, xconst_vel,deltat ;
tr->tstart = t ;
tr->xstart = xstart ;/*depth */
tr->xend = xend ;/* c_depth */
tr->amax = fabs(amax) ;
tr->vmax = fabs(vmax)/2 ;
tr->x = xstart ;
tr->xdot = 0.0;
tr->phase = TR_ACCELERATING;
taccel = tr->vmax/tr->amax;
/* compute the distance traveled if we accelerate to max speed and
then decelerate back to zero speed */
xtriangle = tr->amax*taccel*taccel ;
dx = xend-xstart ;
/*
fprintf(stderr,"INITIALIZATION time = %5.3f  xdot = %5.3f amax = %5.3f,vmax =
```

```
%5.3f\n " ,t,tr->xdot,tr->amax,tr->vmax);
*/
/* test if the needed velocity profile is a trapezoid */
if(fabs(dx) > xtriangle)
{
tr->t1 = tr->tstart + taccel ;
xconst_vel = fabs(dx)-xtriangle ;   /* distance covered at constant vel*/
tr->t2 = tr->t1 + xconst_vel/tr->vmax ;
tr->tend = tr->t2 + taccel ;
tr->type = TR_TRAPEZOIDAL ;
}
else
{
 taccel = sqrt(fabs(dx)/tr->amax) ;
tr->t1=tr->t2 = tr->tstart + taccel ;
tr->tend = tr->t1+taccel ;
tr->type = TR_TRIANGULAR ;
}

}
/*****************************************************************************/
void do_traj_hdg(traj_new *tr, double t, double *px, double *pxdot)
{

double dx, s, dt ;
dx = delta_heading_deg(tr->xend-tr->xstart) ;
dt=.02;
/* s contains the sign of the position change from xstart to xend */
s = 1.0 ;
if(dx < 0.0)s=-1.0 ;
if(t < tr->t1)
{
dt = t - tr->tstart ;
*pxdot = s*tr->amax*dt ;
*px = tr->xstart + s*dista(tr->amax,dt) ;
tr->phase = TR_ACCELERATING ;
}
else
{
if( (t < tr->t2) && (tr->type == TR_TRAPEZOIDAL))
{
dt = t-tr->t1 ;
*pxdot = s*tr->vmax ;
```

```
*px = tr->xstart + s*dista(tr->amax,(tr->t1-tr->tstart)) +
*pxdot*dt ;
tr->phase = TR_CONSTANT_VELOCITY;
}
else
{
if(t < tr->tend)
{
dt = tr->tend - t;
*pxdot = s*tr->amax*dt ;
*px = tr->xend - s*dista(tr->amax,dt) ;
tr->phase = TR_DECELERATING;
}
else
{
 *px = ALPHA_TR*tr->x + BETA_TR*tr->xend ;
  *pxdot = 0.0 ;
 tr->phase = TR_DONE;
}
}
}
tr->x = *px ;
tr->xdot = *pxdot ;
}
/*=====================================================*/
void do_traj_z(traj_new *tr, double t, double *px, double *pxdot)
{

      double dx, s, dt ;
      dx = tr->xend-tr->xstart ;
      dt=.02;
      /* s contains the sign of the position change from xstart to xend */
      s = 1.0 ;
      if(dx < 0.0)s=-1.0 ;
      if(t < tr->t1)
      {
            dt = t - tr->tstart ;
            *pxdot = s*tr->amax*dt ;
            *px = tr->xstart + s*dista(tr->amax,dt) ;
            tr->phase = TR_ACCELERATING ;
      }
      else
      {
```

```
        if( (t < tr->t2) && (tr->type == TR_TRAPEZOIDAL))
        {
            dt = t-tr->t1 ;
            *pxdot = s*tr->vmax ;
            *px = tr->xstart + s*dista(tr->amax,(tr->t1-tr->tstart)) +
                                                *pxdot*dt ;
            tr->phase = TR_CONSTANT_VELOCITY;
        }
        else
        {
            if(t < tr->tend)
            {
                dt = tr->tend - t;
                *pxdot = s*tr->amax*dt ;
                *px = tr->xend - s*dista(tr->amax,dt) ;
                tr->phase = TR_DECELERATING;
            }
            else
            {
                *px = ALPHA_TR*tr->x + BETA_TR*tr->xend ;
                *pxdot = 0.0 ;
                tr->phase = TR_DONE;
            }
        }
    }
    tr->x = *px ;
    tr->xdot = *pxdot ;

}
```

References

[1] J.G. Bellingham. Personal Communication. 1995.

[2] J.G. Bellingham, J.S. Willcox. Optimizing AUV Oceanographic Surveys. Proceedings of the 1996 Symposium on Autonomous Underwater Vehicle Technology, Institute of Electrical and Electronics Engineers, Inc., June 1996.

[3] A.A. Bennett, J.J. Leonard, J.G. Bellingham. Bottom Following for Survey-Class Autonomous Underwater Vehicles. Proceedings of the Ninth International Symposium on Unmanned Untethered Submersible Technology, Autonomous Undersea System Institute, Lee, NH, September 1995.

[4] B. Bourgeois & M. Harris. The NRL "ORCA" Oceanographic Survey Vessel, Proceedings of the Ninth International Symposium on Unmanned Untethered Submersible Technology, Autonomous Undersea System Institute, Lee, NH, September 1995.

[5] T. Curtin, J.G. Bellingham, J. Catipovic, and D. Webb. Autonomous Oceanographic Sampling Networks. *Oceanography,* Vol.6, No.3, 1993.

[6] S.H. Crandall, D.C. Karnopp, E.F. Kurtz, Jr., and D.C. Pridmore-Brown. *Dynamics of Mechanical and Electromechanical Systems*. Robert E. Krieger Publishing Company, Malabar, Florida, 1968.

[7] E.V. Lewis, ed. *Principles of Naval Architecture*, Second revision. Society of Naval Architects and Marine Engineers, Jersey City, New Jersey, 1988.

[8] S.F. Hoerner. *Fluid-Dynamic Drag*. Midland Park, New Jersey,1965.

[9] S.F. Hoerner and H.V. Borst. *Fluid-Dynamic Lift*. Albuquerque, New Mexico, 1985.

[10] F. S. Hover. Physical Modeling of the Odyssey II Vehicle. June 1995.

[11] F. S. Hover. Selection of Physical Parameters for the Odyssey II Vehicle. July 1995.

[12] S.L. Merry, M.J. Large, T.J. Whitten, M.R. Wilkinson, R.J. Babb. Control Surface and Actuator Design for A Low Drag Laminar Flow AUV. Proceedings of the 1996 Symposium on Autonomous Underwater Vehicle Technology, Institute of Electrical and Electronics Engineers, Inc., June 1996.

[13] G.W.K. Moore, Atmospheric Forcing of Deep Convection in the Labrador Sea. January 1996.

[14] J.N. Newman. *Marine Hydrodynamics*. MIT Press, Cambridge, Massachusetts, 1980.

[15] G.K. Pitcher, Solid Lithium Hydride as a Hydrogen Source for Fuel Cells. Proceedings of the 1996 Symposium on Autonomous Underwater Vehicle Technology, Institute of Electrical and Electronics Engineers, Inc., June 1996.

[16] I.J. Potter, G.T. Reader, Design Concepts for AUV Endurance Optimisation, Proceedings of the Ninth International Symposium on Unmanned Untethered Submersible Technology, Autonomous Undersea System Institute, Lee, NH, September 1995.

[17] P.H. Smith, S.D. James, P.B. Keller Development Efforts in Rechargeable Batteries for Underwater Vehicles. Proceedings of the 1996 Symposium on Autonomous Underwater Vehicle Technology, Institute of Electrical and Electronics Engineers, Inc., June 1996.

[18] H. Singh. Personal Communication. 1996.

[19] H. Singh, An Entropic Framework for AUV Sensor Modeling, MIT-WHOI Joint Program Doctoral Thesis, 1995.

[20] H. Singh, D. Yoerger, R. Bachmayer, A. Bradley, W.K. Stewart. Sonar Mapping with the Autonomous Benthic Explorer(ABE). Proceedings of the Ninth International Symposium on Unmanned Untethered Submersible Technology, Autonomous Undersea System Institute, Lee, NH, September 1995.

[21] J.-J. Slotine and W. Li. *Applied Nonlinear Control.* Prentice-Hall, Englewood Cliffs, New Jersey, 1991.

[22] S.M. Smith, K. Heeb, N. Frolund & T. Pantelakis. The Ocean Explorer AUV: A Modular Platform for Coastal Oceanography. Proceedings of the Ninth International Symposium on Unmanned Untethered Submersible Technology, Autonomous Undersea System Institute, Lee, NH, September 1995.

[23] J.-J. E. Slotine. "Tracking Control of Nonlinear Systems Using Sliding Surfaces". MIT Doctoral Thesis, 1983.

[24] J.H. Stannard, D. Stockburger, J. Tregenza, Oxygen Storage and Supply System Tradeoff Study For UUV Propulsion by Aluminum Semi-Fuel Cell, Proceedings of the Ninth International Symposium on Unmanned Untethered Submersible Technology, Autonomous Undersea System Institute, Lee, NH, September 1995.

[25] R. Uhrich & J. Walton, Supervisory Control of Untethered Undersea Systems: A New Paradigm Verified, Proceedings of the Ninth International Symposium on Unmanned Untethered Submersible Technology, Autonomous Undersea System Institute, Lee, NH, September 1995.

[26] C. von Alt, B. Allen, T. Austin, R. Stokey, Remote Environmental Measuring Units, Proceedings of the 1994 Symposium on Autonomous Underwater Vehicle Technology, Cambridge, MA, July 1994.

[27] D.R. Yoerger, J.G. Cooke, and J.-J. Slotine. The influence of thruster dynamics on underwater vehicle behavior and their incorporation into control system design. *IEEE Journal of Oceanic Engineering*, 15:167-178, 1990.

[28] G. Zorpette, Autopilots of the Deep, IEEE Spectrum,vol.31,No.8, pp.38-44, Aug 1994

[29] G. Zorpette, Power Sources for the Long Haul, IEEE Spectrum,vol.31,No.8, p.41, Aug 1994

3 483NPG TH 2665
10/99 22527-200

DUDLEY KNOX LIBRARY
3 2768 00366838 5